

The image is a complex collage. At the top right is a calendar for December 1981, showing the 19th as a Friday. Below the calendar is a newspaper clipping from 'Sde Jaargang nr. 19' in Netherland, dated December 19, 1981. The headline reads 'ROBBERIJ IN NEDERLAND' (Robbery in Netherland) and reports on a robbery at the 'KIM GEBRUIKERS CLUB' where a man was injured and a large sum of money was stolen. The bottom half of the image is dominated by a technical document, likely a program listing or assembly code. The code is written in a monospaced font and includes instructions like 'LDI \$00', 'STAY OFFRND', 'ENDDO', 'DEX', 'IN', 'BNE L0006A', 'RTS', 'LDI \$00', 'STAY OFFRND', 'ENDDO', 'DEX', 'IN', 'BNE L0006A', 'RTS', 'LDI \$00', 'STAY OFFRND', 'ENDDO', 'DEX', 'IN', 'BNE L0006A', 'RTS'. The code is organized into columns, with some lines starting with 'KENDNER'. In the center of the collage is a 3D isometric drawing of a box with the number '6502' on its front face. The box is rendered with perspective, showing its top, front, and right sides. The background of the entire collage is a light gray with a subtle grid pattern.

61



Het doel van de vereniging is het bevorderen van de kennis uitwisseling tussen de gebruikers van 6502 computers zoals KIM, SYM, JUNIOR, AIM65, System-65, Acorn, PET en CBM, Apple, Atari, ITT2020, PC100, Ohio Scientific Challengers, etc. Dit doel wordt onder andere gerealiseerd door vijf maal per jaar een clubbijeenkomst te houden op de derde zaterdag van oneven maanden uitgezonderd de maand juli. Ook wordt vijf maal per jaar het huisorgaan "DE 6502 KENNER" gepubliceerd, welke verschijnt op de derde zaterdag van de maanden februari, mei, augustus, oktober en december.

Naast deze vaste activiteiten kunnen de leden gebruik maken van club faciliteiten zoals de KIM-CLUB-KIM, de KIM-CLUB-JUNIOR en de cassette bibliotheek.

De KIM gebruikers club Nederland is een volledig onafhankelijke vereniging met statuten en een bestuur. De club is ingeschreven bij de Kamer van Koophandel en Fabrieken voor Hollands Noorderkwartier te Alkmaar onder nummer 634305

De samenstelling van het bestuur is thans als volgt :

Dagelijks bestuur	
Voorzitter	Anton Müller
Secretaris	Ruud Uphoff
Penningmeester	Ted Schouten
Accomodate	Riche van Steen
Hardware	Peter Visser
Ledenadministratie	Bob van de Oudewetering
Organisatie	Rinus Vleesch Dubois
Redactie	Hans Otten
Software	Sebo Woldringh
KIM GEBRUIKERS CLUB NEDERLAND	
Voorburgpad 10	
6843 EM ARNHEM	

Penningmeester : giro nummer 3757649 tnv T.Schouten Junopits 57 Haarlem
tel. 085 - 816935 op werkdagen van 19.00 tot 20.00
f 40,- per kalenderjaar

INHOUDSOPGAVE 6502 KENNER nr 19 DECEMBER 1981

1. Inhoudsopgave
2. Van het bestuur, Anton Müller
3. Van de redactie, Hans Otten
4. UITNODIGING CLUB BIJZEEENKOMST
6. Schijf- en leesroutine voor de JUNIOR, Dick Blok
15. Rapportcijfer, Rein Duyts
22. Display op oscilloscoop, C. Totté
28. Aanpassingen aan de Macro-Assembler van C.W. Moser, Van Nieuwenhove Koen
34. Patches op BASIC, Hans Otten
35. Lichtshow voor de JUNIOR en de KIM, K. Kikke
38. 4 kolom printer, A.S. Hankel
40. Printer routine voor H14, Paul de Beer en Hans Otten
43. BREAK voor JUNIOR en KIM, Frans Mepschen
46. Hex dump, Frans Smeehuizen
48. Cassette bibliotheek
49. Vraag en Aanbod, Agenda

De 6502 KENNER is een uitgave van de KIM Gebruikers club Nederland.

Adres voor het in-zenden van en re-akties op artikelen voor de 6502 KENNER: Redaktiesekretaris Hans Otten

Ottoborrengeod 33 3871 MJ HOEVELAKEN

Redactie 6502 KENNER: Anton Müller Hans Otten Willem van Pelt

Geheel of gedeelte-lijke overname van de inhoud van de 6502 KENNER zonder toestemming van het bestuur is verboden. Toepassen van gepu-bliceerde programma's, hardware etc. is al-leen voor persoonlijk gebruik toegestaan.

COPYRIGHT © 1981 KIM Gebruikers club Nederland.

Verschijnt vijf maal per jaar.

Anton Müller
Voorzitter

Deze keer wil het eens hebben over een activiteit die U misschien best aanspreekt. Een paar weken geleden kreeg ik een telefoontje van John van Sprang uit Krimpen aan de IJssel met het verzoek of wij hem wilden steunen met het opstarten van lokale club bijeenkomsten in Den Haag. Na overleg met het dagelijks bestuur is hieruit het volgende naar voren gekomen :

Er wordt een up to date ledenlijst versprekt aan de initiatiefnemers : de kosten voor het produceren en verzenden van de uitnodiging voor de eerste bijeenkomst komen voor rekening van de KIM club, alsmede de zaalhuur voor de eerste bijeenkomst, een en ander binnen zekere voor het bestuur aanvaardbare grenzen. De eerste bijeenkomst in Den Haag is inmiddels geweest en was een geweldig succes. De opkomst was 130 %, dat wil zeggen er werden 20 mensen uitgenodigd en er kwamen er 26. Er was van te voren geen programma vastgesteld. Naar ik heb vernomen komen allen volgendekeer weer.

Voelt U zelf iets voor het starten van lokale of regionale bijeenkomsten neem dan eens contact op met ondergetekende of de secretaris. Het is overigens wel de bedoeling dat qua financieën dergelijke bijeenkomsten zichzelf bedruipen.

Dan nu even iets anders . Wij kunnen voor de redactie nog steeds medewerkers gebruiken die de beschikking hebben over een ontwikkel systeem met Micro Ade en/of ASM/TED van C.W. Moser voor het vertalen van de assembler sources en het produceren van listings op papier. Hoe meer wij het werk voor de 6502 KENNER kunnen verdelen hoe beter en des te meer aandacht kan er aan elk programma worden besteed, hetgeen de kwaliteit van de 6502 KENNER alleen maar ten goede kan komen.

Wat U minimaal nodig heeft is een systeem met 8K RAM en een beeldscherm of toetsenbord met video interface of iets dergelijks en de mogelijkheid om cassettes aan te kunnen maken en enige kennis van de 6502 assembly language. De rest komt vanzelf. Als we zo'n stuk of vijf medewerkers kunnen krijgen dan blijven de werkzaamheden beperkt tot zo'n 50 uur per jaar per medewerker en dat is toch niet teveel gevraagd. Ook kunnen we in de redactie nog wel mensen gebruiken van een kladjé of iets dergelijks een behoorlijk hardware schema kunnen tekenen, hetgeen zo nu en dan eens voorkomt. Heeft U interesse neem dan eens even contact op met de secretaris of ondergetekende. Behalve dat het nuttig werk voor de club is steek je er zelf nog wat van op in de vorm van contacten en kennis. Bovendien helpt je mee aan het in stand houden van een vereniging van zo'n ruim 400 leden ; een vereniging waar je graag lid van zou worden en dat waarschijnlijk om het clubblad : DE 6502 KENNER.

6502 KENNER 19 ligt hier voor U, het laatste nummer in 1981

en het laatste nummer waarin U mijn naam als medewerker zal

tegenkomen aan de 6502 KENNER en het bestuur van de KIM club.

Twee jaar lang heb ik mogen helpen met de club activiteiten

en in die twee jaar de club wat leden aantal betreft zien

verdreievoudigen zonder dat het eind van die groei in zicht is

gelukkig. Aan die groei heb ik wat geholpen door wat contacten

naar buiten te leggen voor de club maar drie personen wil ik

noemen die het gepresteerd hebben alles in goede administratie

tieve banen te leiden : Anton Müller, secretaris in 1980 en

nu voorzitter en Ruud Uphoff, de huidige secretaris, de

financiën werden uitstekkend beheerd door Ted Schouten. Deze

personen zijn van onschatbare betekenis geweest voor de club.

Zelf stop ik nu met actieve deelneming aan het club gebeuren

gedwongen door mijn drukke werkring en de activiteiten die

ik als redacteur van Radio Bulletin ontlooi. Wel hoop ik nog

viele jaren als lid van de club samen met U onze hobby en ook

mijn vak nu te mogen beoefenen.

Een van de fijnste ervaringen in de KIM club is geweest samen

met mede hobbyisten te kunnen werken aan de 6502 soft- en hard-

ware op een gelijke basis. Alleen doen we ons werk voor de club

in onze vrije tijd. Het clubblad wordt ook zonder enige commercie-

le binding of oogmerk gepubliceerd. We laten daarbij regel-

matig een steek vallen maar hopen dat U dat ons niet kwalijk

neemt. De 6502 KENNER is een blad gewijd aan de 6502 voor en

door leden van de KIM club. De leden dragen zelf de artikelen etc

aan waarmee het clubblad gevuld kan worden. Het resultaat mag

gerust uniek worden genoemd, in Nederland maar ook internationaal.

Voor een gevaar wil ik nog waarschuwen. De meeste beginnende

computer hobbyisten denken op een volledig verkeerde manier

over hobby computers. Hobby of personal computers zijn namelijk

gewone computers en het schrijven van software verschilt nauwelijks

met dat voor mini en mainframe computers. Luister en lees daarom

de ervaringen van ervaren programmeurs en hun ideeën over ge-

structureerd programmeren : Vind niet opnieuw het wiel uit !

Tot ziens.

Hans Otten

UITNODIGING KIM CLUB BIJ EENKOMST
=====

Locatie : Forbo Krommenie
Krommenie - Assendelft
Tijdstip : 16 januari 1982 10.00 tot 17.00 uur

Kosten : f 20,- inclusief koffie, lunch en frisdrankje
Route : zie ommezijde

Programma :

9.30 Ontvangst met koffie .

10.00 Opening , welkomswoord door voorzitter en gastheer.

10.15 Lezing door Ted Schouten.

Onderwerp : Temperatuurmeting mbc AIM65.

11.00 Koffie pauze

11.15 Lezing door Hans Otten.

Onderwerp : De 6809 microprocessor hard en software.

Programmeer model en instructieset en belang

daarvan voor reentrant position indepent programme-

ren , het OS/9 operating system

12.00 Lunchpauze

12.45 Deel 2 van de lezing over de 6809 door Hans Otten

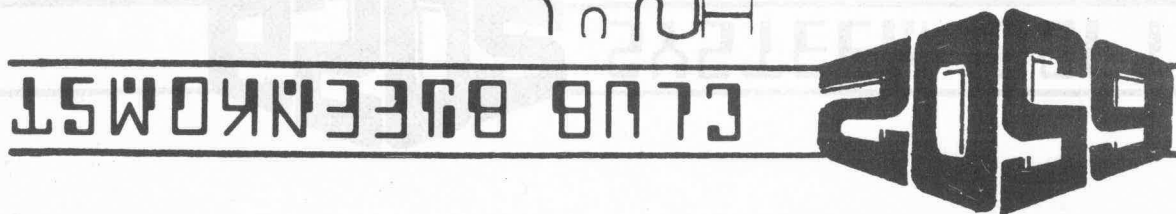
13.30 Inleiding gestructureerd programmeren met voorbeelden in

Basic en Assembler door Fred Harthoorn.

14.15 Theepauze, forum en markt

15.00 Behandeling instructie set van de 6502

15.30 tot 17.00 Informele discussies demonstraties etc



6502 SYSTEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 01

0010: 0200

ORG \$0200

SCHRIF- EN LEESROUTINE VOOR DE JUNIOR

AUTEUR: DICK BLOK

..... RIJKSSTRAATWEG 594 2H
2026 RD HAARLEM

COPYRIGHT (C) BY DICK BLOK

Met dank aan Sebó Woldringh voor zijn adviezen.

In KIM-Kenner 14 van december 1980 werden een schrift- en een leesroutine voor de veelbesproken Kansas City interface van M.C.P. (Nu CHIP) gepubliceerd door Sebó Woldringh. Daarmee heeft hij voorzien in een grote behoefte, al ontbraken op dat moment enige kwaliteitsaspecten. Om hieraan tegemoet te komen, speciaal met het oog op diegenen die nog niet tot de aanschaf van de Elektuur-interfacekaart (kunnen) overgaan, is deze routine ontwikkeld. Diegenen die inmiddels over deElektuur-interfacekaart beschikken zullen veel herkenningspunten ontdekken. (Bronvermelding: zie einde lijsting).

Het gebruik van de geheugenruimte, de adressen 0200 - 03E5, veronderstelt wel een geheugenuitbreiding.

Het programma wordt als volgt op cassette gezet: eerst 256 synchronisatie tekens (hex. 1B), vervolgens een programma-nummer (identificatienummer: ID), daarna start- en eindadres en tenslotte het eigenlijke programma, gevolgd door 2 bytes checksum.

Met terugkijken wordt eerst gekeken naar de synchro- nisatie tekens, daarna zoekt de Junior het begin van de data, vervolgens wordt het ID van de band gehaald en vergeleken met de opgegeven ID, 01 t/m FE. Wordt ID=00 opgegeven, dan leest de Junior het eerst op de band volgende programma in. Indien ID=FF, dan leest de Junior alleen het programma in zonder ID en start/eindadres van de band over te nemen. Wel moet dan eerst een nieuw start- en eindadres opgegeven worden.

Er is ook een ingebouwde controle op fouten bij het inlezen van de gegevens van de band naar de computer. Dit gaat als volgt:

Tijdens het schrijven op de band wordt alle data bij

DE 6502 KENNER

6

0500:
0490:
0480:
0470:
0460:
0450:
0440:
0430:
0420:
0410:
0400:
0390:
0380:
0370:
0360:
0350:
0340:
0330:
0320:
0310:
0300:
0290:
0280:
0270:
0260:
0250:
0240:
0230:
0220:
0210:
0200:
0190:
0180:
0170:
0160:
0150:
0140:
0130:
0120:
0110:
0100:
0090:
0080:
0070:
0060:
0050:
0040:
0030:
0020:
0010:

010:
020:
030:
040:
050:
060:
070:
080:
090:
100:
110:
120:
130:
140:
150:
160:
170:
180:
190:
200:
210:
220:
230:
240:
250:
260:
270:
280:
290:
300:
310:
320:
330:
340:
350:
360:
370:
380:
390:
400:
410:
420:
430:
440:
450:
460:
470:
480:
490:
500:
510:
520:
530:
540:
550:
560:
570:
580:
590:
600:
610:
620:
630:
640:
650:
660:
670:
680:
690:
700:
710:
720:
730:
740:
750:
760:
770:
780:
790:
800:
810:
820:
830:
840:
850:
860:
870:
880:
890:
900:
910:
920:
930:
940:
950:
960:
970:
980:
990:
1000:

elkaar opgesteld in een veld van 2 bytes en wordt be-
waard op adres 1A73 en 1A74, respectievelijk CHKL en
CHKH. Dit noemen we de checksum (controlegetal).
Bij het teruglezen wordt dit ook gedaan. Aan het eind
van het programma staat de checksum op de band. Als
alles is ingelezen wordt de checksum op de band verge-
leken met de berekende checksum. Stemt dit overeen,
dan wordt teruggeproven naar de resetroutine en
op het display verschijnt, in tegenstelling tot het
eindadres van de elektuuroutine, het startadres van
het zoutst ingelezen programma. Er kan dan meteen
worden gereset.
Gaat er iets fout met het van de band lezen, dan wordt
teruggeproven naar de monitor en verschijnt het
startadres van het leesprogramma op het display. Men
kan het kiezen dan opnieuw proberen.

De volgende adressen zijn belangrijk om te onthouden:

SAL = 1A75 (Start adres laag)
SAH = 1A76 (Start adres hoog)
EARL = 1A77 (Eind adres laag)
EAH = 1A78 (Eind adres hoog)
ID = 1A79 (Identificatie nummer)

0200 = Start address schrijffprogramma.
02AC = Start address leesprogramma.

Zet men de routines in een andere geheugenruimte, dan
moeten ook de adressen 0362 en 0365 worden aangepast.
De opgegeven versie van het programma werkt op onze
veel 100 Baud, zoals het programma uit KIM-Kenner 14.
Voor wijziging in 300 Baud zie het commentaar
bij de equates en timer.
Voor wijzigingen aan de cassette-interface en het ge-
bruik van een voorversterker, zie "Ervaringen met
de Junior" van Wim van Peit en John van Sprang.

Schrijven gebeurt via PBS.
Lezen gebeurt via PBB.

JUNIOR MONITOR DEFINITIES:

PAD	* \$1A80	PIA A data register
PADD	* +01	PIA A data direction register
PBD	* +01	PIA B data register
PBDD	* +01	PIA B data direction register
TIMER	* \$1A77	Timer word 100 Baud.
(Deze locatie wijzigen in 1A76 voor 300 Baud).		
RESET	* \$1C1D	Reset routine
SHOW	* \$1DCC	7 segment display routine

PAGINA NUL LOCATIES:

TEMP	* \$00E0	
INL	* \$00F8	
INH	* +01	INL
POINTL	* +01	INH
POINTH	* +01	POINTL
TEMPX	* +02	POINTH

VASTE LOCATIES IN PAGINA 1A:

CHKL	* \$1A73	CHKL
CHKH	* +01	CHKL
SAL	* +01	CHKH
SAH	* +01	SAL
EAL	* +01	SAH
EAL	* +01	EAL
EAH	* +01	EAL
ID	* +01	EAH

EQUATES:

HALF	* \$0005	Half bit tijd Pij 100 Baud
(Wijzigen in \$001A Pij 300 Baud).		
HEEL	* \$000A	Hele bit tijd Pij 100 Baud
(Wijzigen in \$0034 Pij 300 Baud).		
TWEEH	* \$001B	2 1/2 bit tijd Pij 100 Baud
(Wijzigen in \$007E Pij 300 Baud).		

6502 SYSTEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 04

SCHRIJFROUTINE:

540: 0200 A9 20	WRITE	LDAIM #20	Zet data direction register
550: 0202 8D 83 1A	STA	PBDD	PBS op output
560: 0205 A9 00	LDAIM #00		Maak inhoud checksum schoon
570: 0207 8D 73 1A	STA	CHKL	
580: 020A 8D 74 1A	STA	CHKH	
590: 020D A2 FF	LDXIM #FF		Schrijf 256 SYNC tekens weg
600:			
610: 020F A9 16	MSYNC	LDAIM #16	(Synchronisatie teken)
620: 0211 86 FD	STX	TEMPX	
630: 0213 20 6F 02	JSR	OBYTE	
640: 0216 A6 FD	LDX	TEMPX	
650: 0218 CA	DEX		
660: 0219 D0 F4	BNE	MSYNC	
670: 021B A9 2A	LDAIM #2A		Schrijf start data teken weg
680: 021D 20 6F 02	JSR	OBYTE	
690: 0220 AD 79 1A	LDA	ID	Schrijf identificatie nummer weg
700: 0223 20 6F 02	JSR	OBYTE	
710: 0226 AD 75 1A	LDA	SAL	
720: 0229 85 F8	STA	INL	Schrijf start adres weg
730: 022B 20 6C 02	JSR	OCHKT	
740: 022E AD 76 1A	LDA	SAH	
750: 0231 85 F9	STA	INH	
760: 0233 20 6C 02	JSR	OCHKT	
770: 0236 AD 77 1A	LDA	EAL	
780: 0239 20 6C 02	JSR	OCHKT	
790: 023C AD 78 1A	LDA	EAH	
800: 023F 20 6C 02	JSR	OCHKT	
810:			
820: 0242 A0 00	WDATA	LDAIM #00	Schrijf data weg
830: 0244 B1 F8	LDAIM	INL	
840: 0246 20 6C 02	JSR	OCHKT	
850: 0249 E6 F8	INC	INL	Verhoog start adres met 1
860: 024B D0 02	BNE	WDATA	
870: 024D E6 F9	INC	INH	
880:			
890: 024F A5 F8	WDATA	LDA	Indien startadres = eindadres + 1
900: 0251 CD 77 1A	CMP	EAL	dan berekende checksum op tape
910: 0254 D0 EC	BNE	WDATA	zetten en terug naar reset.
920: 0256 A5 F9	LDA	INH	
930: 0258 CD 78 1A	CMP	EAH	
940: 025B D0 E5	BNE	WDATA	
950: 025D AD 73 1A	LDA	CHKL	
960: 0260 20 6F 02	JSR	OBYTE	
970: 0263 AD 74 1A	LDA	CHKH	
980: 0266 20 6F 02	JSR	OBYTE	
990: 0269 4C 1D 1C	JMP	RESET	

DE 6502 KENNER

2010:	026C	20	9C	02	OCHKT	JSR	CHKT	Weschrijven met checksum
2020:	026F	18			OBYTE	CLC		Weschrijven zonder checksum
2030:	0270	48			PHA			Stop te schrijven byte op de stack
2040:	0271	20	89	02	JSR	OUTBI		en geeft start bit
2050:	0274	68			PLA			Haal te schrijven byte op
2060:	0275	80	08		LDYIM \$08			Aantal bits per byte in Y
2070:	0277	0A			ASLA			Schuit bit in carry flag
2080:	0278	48			PHA			Restant op stack
2090:	0279	20	89	02	JSR	OUTBI		Schuit een bit weg
2100:	027C	68			PLA			Haal resterende bits weest op
2110:	027D	88			DEY			Herhaal dit tot dat
2120:	027E	D0	F7		BNE	OBTEA		We 8 bits hebben gehad.
2130:	0280	38			SEC			Schuit 2 stop bits
2140:	0281	20	89	02	JSR	OUTBI		
2150:	0284	38			SEC			
2160:	0285	20	89	02	JSR	OUTBI		
2170:	0288	60			RTS			Ga terug naar hoofd routine
2180:	0289	A2	00		LDXIM \$00			Schuit een bit
2190:	028B	90	01		BCC	OUTA		naar PBS
2200:	028D	CA			DEX			
2210:	028E	9E	82	1A	STX	PBD		Zet hele bit tijd
2220:	0291	A9	0A		LDXIM HEEL			
2230:	0293	8D	F7	1A	STA	TIMER		voor timer en
2240:	0296	2C	F7	1A	BIT	TIMER		laat timer aflopen
2250:	0299	10	FB		BPL	OUTB		
2260:	029B	60			RTS			Ga terug naar hoofd routine
2270:	029C	A8			TAY			Bereken de check sum
2280:	029D	18			CLC			
2290:	029E	6D	73	1A	ADC	CHKL		
2300:	02A1	8D	73	1A	STA	CHKL		
2310:	02A4	AD	74	1A	LDA	CHKH		
2320:	02A7	69	00		ADDCIM \$00			
2330:	02A9	8D	74	1A	STA	CHKH		
2340:	02AC	98			TYA			
2350:	02AD	60			RTS			
2360:	02A7	69	00		ADDCIM \$00			
2370:	02A9	8D	74	1A	STA	CHKH		
2380:	02AC	98			TYA			
2390:	02AD	60			RTS			
2400:								
2410:								
2420:								
2430:								
2440:	02AE	A9	00		READ	LDXIM \$00		Zet data direction register voor input
2450:	02B0	8D	83	1A	STA	PBDD		
2460:	02B3	8D	73	1A	STA	CHKL		Maak checksum schoon
2470:	02B6	8D	74	1A	STA	CHKH		
2480:	02B9	20	81	03	WAIT	RDPBD		Wacht tot er wat komt
2490:	02BC	D0	FB		BNE	WAIT		
2500:	02BE	20	87	03	SYNCA	JSR	RBYTE	Lees een byte en

LEESROUTINE:

6502 SYSTEM SOFTWARE

KIM SOFTWARE LIBRARY

PAGE 06

2510: 02C1 C9 16	CMPIM #16	Klik of het een SYNC is.
2520: 02C3 D0 F9	BNE SYNCA	
2530: 02C5 A2 0A	LDXIM \$0A	Lees 10 SYNC tekens
2540: 02C7 20 87	JSR RBYTE	
2550: 02CA C9 16	CMPIM #16	
2560: 02CC D0 F0	BNE SYNCA	
2570: 02CE CA	DEX	
2580: 02CF D0 F6	BNE SYNCB	
2590: 02D1 20 87	JSR RBYTE	Wacht op data begin teken
2600: 02D4 C9 2A	CMPIM #2A	
2610: 02D6 F0 06	BEG LDID	
2620: 02D8 C9 16	CMPIM #16	
2630: 02DA D0 E2	BNE SYNCA	
2640: 02DC F0 F3	BEG DBGIN	
2650: 02DE 20 87	JSR RBYTE	Lees identifikatie nummer
2660: 02E1 CD 79	CMP ID	
2670: 02E4 F0 25	BEG LDSA	Indien gelijk dan naar laadroutine
2680: 02E6 AD 79	LDA ID	
2690: 02E9 C9 00	CMPIM #00	Indien 00 ook naar laadroutine
2700: 02EB F0 1E	BEG LDSA	
2710: 02ED C9 FF	CMPIM \$FF	Indien FF dan slaan we behalve de ID
2720: 02EF F0 02	BEG IGRE	ook het start- en eind adres over
2730: 02F1 D0 BB	BNE READ	Zoek naar volgende file.
2740: 02F3 A2 04	LDXIM \$04	
2750: 02F5 20 87	JSR RBYTE	
2760: 02F8 20 9C	JSR CHKT	
2770: 02FB CA	DEX	
2780: 02FC D0 F7	BNE IGRA	
2790: 02FE AD 75	LDA SAL	Haal nieuw start- en eindadres op
2800: 0301 85 F8	STA INL	en zet die op de juiste plaats
2810: 0303 AD 76	LDA SAH	
2820: 0306 85 F9	STA INH	
2830: 0308 4C 33	JMP RDATA	Ga data lezen.
2840: 030B 20 87	JSR RBYTE	Laad start adres van tape met checksum
2850: 030E 20 9C	JSR CHKT	berekening
2860: 0311 85 F8	STA INL	
2870: 0313 8D 75	STA SAL	
2880: 0316 20 87	JSR RBYTE	
2890: 0319 20 9C	JSR CHKT	
2900: 031C 85 F9	STA INH	
2910: 031E 8D 76	STA SAH	
2920: 0321 20 87	JSR RBYTE	Laad eind adres van tape met checksum
2930: 0324 20 9C	JSR CHKT	berekening
2940: 0327 8D 77	STA EARL	
2950: 032A 20 87	JSR RBYTE	
2960: 032D 20 9C	JSR CHKT	
2970: 0330 8D 78	STA EAH	
2980: 0333 20 87	JSR RBYTE	Lees data van tape met checksum-
2990: 0336 20 9C	JSR CHKT	berekening
3000: 0339 A0 00	LDYIM \$00	

DE 6502 KENNER

3010: 033B 91 F8	STAY INL	INL	Store character op seindexeerde adres
3030: 033F D0 02	BNE	RDTAA	Verhoor adres met 1
3040: 0341 E6 F9	INC	INH	
3050: 0343 A5 F8	LDA	INH	
3060: 0345 CD 77 1A	CMP	INH	Als het startadres gelijk is aan het eindadres + 1 dan zijn we klaar.
3070: 0348 D0 E9	BNE	RDATA	
3080: 034A A5 F9	LDA	INH	
3090: 034C CD 78 1A	CMP	EAH	
3100: 034F D0 E2	BNE	RDATA	
3110: 0351 20 87 03	JSR	RBYTE	Lees checksum van tape en
3120: 0354 CD 73 1A	CMP	CHKL	vergelijk die met berekende checksum
3130: 0357 D0 08	BNE	ERROR	
3140: 0359 20 87 03	JSR	RBYTE	
3150: 035C CD 74 1A	CMP	CHKH	
3160: 035F F0 08	BEQ	END	
3170: 0361 A9 AE	LDAIM	READ	
3180: 0363 85 FA	STA	POINTL	Als checksum fout is,
3190: 0365 A9 02	LDAIM	READ	dan terug met begin adres leestrouine.
3200: 0367 85 FB	STA	POINTH	
3210: 0369 4C 1D 1C	JMP	RESET	
3220: 036C AD 75 1A	LDA	SAL	Als de checksum correct is,
3230: 036F 85 FA	STA	POINTL	dan terug met beginadres van
3240: 0371 AD 76 1A	LDA	SAH	gelezen programma.
3250: 0374 85 FB	STA	POINTH	
3260: 0376 4C 1D 1C	JMP	RESET	
3270: 0379 8D F7 1A	STA	TIMER	Timer tijd
3280: 037C 2C F7 1A	BIT	TIMER	Laat timer aflopen.
3290: 037F 10 F8	BPL	DELAY	
3300: 0381 AD 82 1A	LDA	PBD	Lees PIA B PBE
3310: 0384 29 40	ANDIM	\$40	Zet ongewenste bits uit
3320: 0386 60	RTS		Terug naar caller
3330: 0387 20 81 03	JSR	RDPBD	Machtius voor start bit
3340: 038A D0 FB	BNE	RBYTE	
3350: 038C 20 81 03	JSR	RDPBD	
3360: 038F D0 FB	BNE	RBYTE	
3370: 0391 A9 05	LDAIM	HALF	Zet halve bit tijd
3380: 0393 20 79 03	JSR	DELAY	
3390: 0396 A0 08	LDAIM	\$08	Antal bits per byte op de stack
3400: 0398 48	PHA		
3410: 0399 A9 0A	LDAIM	HEEL	Zet hele bit tijd; haal een bit.
3420: 039B 20 79 03	JSR	DELAY	
3430: 039E 18	CLC		Zet carryflag indien "1".
3440: 039F F0 01	BEQ	RBTB	
3450: 03A1 38	SEC		
3460: 03A2 68	PLA		
3470: 03A3 2A	ROLA		
3480: 03A4 48	PHA		
3490: 03A5 88	DEY		
3500: 03A6 D0 F1	BNE	RBTB	Verlaas loop delimeter
			en herhaal dit tot dat 8 bits verwerkt

3510:	03A8 68	PLA	Red accumulator in temporary storage.
3520:	03A9 85 E0	STA	
3530:	03AB A9 18	LDAIM	TWEEH
3540:	03AD 20 C2 03	JSR	VUTAP
3550:	03B0 A5 E0	LDA	TEMP
3560:	03B2 60	RTS	
3570:	03B3 20 87 03	JSR	RBYTE
3580:	03B6 C9 16	CMPIM	#16
3590:	03B8 F0 07	BEG	UIT
3600:	03BA A9 0A	LDAIM	HEEL
3610:	03BC 20 79 03	JSR	DELAY
3620:	03BF 10 F2	BPL	BEGIN
3630:	03C1 60	RTS	
3640:	03C2 8D F7 1A	STA	VUTAP
3650:	03C5 86 FD	STX	TEMPX
3660:	03C7 A9 BF	LDAIM	#BF
3670:	03C9 8D 83 1A	STA	PBDD
3680:	03CC A9 7F	LDAIM	\$7F
3690:	03CE 8D 81 1A	STA	PADD
3700:	03D1 A0 04	LDAIM	\$04
3710:	03D3 A2 10	LDAIM	\$10
3720:	03D5 A5 E0	LDA	TEMP
3730:	03D7 20 CC 1D	JSR	SHOW
3740:	03DA 88	DEY	
3750:	03DB D0 FE	BNE	DISPL
3760:	03DD A6 FD	LDX	TEMPX
3770:	03DF 2C F7 1A	BIT	WACHT
3780:	03E2 10 FB	BPL	
3790:	03E4 60	RTS	
3800:			
3810:			
3820:			
3830:			
3840:			
3850:			
3860:			
3870:			
3880:			
3890:			
3900:			
3910:			
3920:			
3930:			

Bronnen:

- 1) Lees- en schrijfroutine van Sebo Moldring in KIM-kenner 14.
- 2) Programma SUPERTAPE van Y.L. Bicknese in KIM-kenner 12
- 3) Junior computer boek, deel 1 en 2 van Elekture
- 4) Microcomputer systems van Camp, Smay en Triska
- 5) KIM-1 Monitor programma (Mos Technology)
- 6) Programma VUTAPE uit First Book of KIM.

lend een nul en een.

de LED constant te laten branden.

0020 : 0100

0020\$ 940

0050
0670
0870
0270
0370
0570
0770
0270
0270
0170
0070
0630
0830
0230
0930
0350
0740
0330
0320
0310
0030
0620
0820
0220
0260
0250
0720
0330
0220
0120
0020
0610
0180
0710
0160
0150
0140
0130
0210
0110
0010
0090
0080
0070
0060
0050
0050
0040
0030
0020
0010

*P13126324

1214 21JPMZ EP 1P1OM '51 0 J21C13 U22 51H

* (521 JPMZ 2P UPA JPP101 124

7 DISPLAYS : NET 96MIBS: REPLYED: RECHT 7 DISPLAYS

U.S. DEPARTMENT OF JUSTICE

THIS DE TABLE LABELS, IS, CONTAINED IN THE

Transmitter, Dan, et al. on yet Volgende cijfer.

Linker 4 displays: cluster, rechter 2 displays

starten, kan men het in cijfer intikken,

Het programma start op adres 0200. Na het

1 aatsts Zwaarte most 00 OPRES/AGEN MORDEN.

gen op de plaatsen 0000, 0001 e.v. Na de

de bepallen. De zwaartes moeten worden opgesla-

Met dit programma kan je een gewogen gemiddelde -

Het profamma "Kappott" is:

1145).

decimaal punt (5 twee naar links, 55 een, 679

It is seen from the above that the

(-7+) ! n een keer ! investikt kan worden en dat

Gemaakt bijk is ook, dat de min of de plus (van

VOOR NAAR ACTIEN OF ANDERSON DOORLOOPEN.

programma kan dat wel, je kunt de lijst van

tickets just differs niet controleren. Bij dit

Kent een belangrijke bezwaar: de Kunt de 1792-

tekenmachine is nog niet zo gemakkelijk en

SC4111 and PU211M24. Het u!trekenen W.B.V. een

—128 uae 1952aw stat!D ap uaq94 stat!D4202

Bij het opstellen van rapport- of schoolonder-

*B U I P I A I U T

0872 - 68220 : Tel

1619 BV ANDICK

2 62M001G 72E16H1D

Rein Dvts

Programma: Kappottciufet

APP LIC AT I E

2099

KIM SOFTWARE LIBRARY 3045

PAGE 01 APR 81 0901Z APR 81

```

0510:
0520: Toetsfuncties:
-----
0530:
0540:
0550: + : Omhoog in tabel
0560: DA : Omlaag in tabel
0570: AD : Clear
0580: PC : Spring naar het begin v
0590: F : Spring naar het begin v
0600: een nieuwe serie
0610: C : -, trekt 0,25 van het c
0620: D : +, telt 0,25 bij het c
0630: R : Wordt gebruikt voor het
0640:
0650: Page zero locaties:
-----
0660:
0670:
0680: ZWARR * : $0000 Zwaartes
0690: KEY * : $0049
0700: TEMPA * :
0710: TEMPB * :
0720: TEMPC * :
0730: TEMPD * :
0740: LABEL * : $0050
0750: INH * : $00F9
0760: POINTL * : INH +01
0770: POINTH * : POINTL +01
0780:
0790: Junior monitor routines:
-----
0800:
0810:
0820: SCANDS * : $1DBE
0830: GETKEY * : $1DF9
0840:
0850: 20 01 03 MAIN JSR INIT
0860: 96 49 STX KEY
0870: 20 37 JSR KEYDIS
0880: 96 49 LDX KEY
0890: C9 12 CMPIM $12 +
0900: D0 06 BNE MAINB
0910: 20 AB JSR PLUS
0920: 4C 03 JMP MAINA
0930: C9 11 CMPIM $11 DA
0940: D0 06 BNE MAINC
0950: 20 C0 JSR MIN
0960: 4C 03 JMP MAINA
0970: C9 10 CMPIM $10 AD
0980: D0 06 BNE MAIND
0990: 20 89 JSR CLEAR
1000: 4C 03 JMP MAINA

```



```

1010: 0228 C9 14      MAIND  CMPIM $14      'PC'
1020: 022A D0 06      BNE      MAINE
1030: 022C 20 27      JSR      LYST
1040: 022F 4C 03      JMP      MAINA
1050: 0232 C9 0F      MAINE  CMPIM $0F      'F'
1060: 0234 D0 03      BNE      MAINF
1070: 0236 4C 00      JMP      MAIN
1080: 0239 C9 0C      MAINF  CMPIM $0C      'C'
1090: 023B D0 06      BNE      MAING
1100: 023D 20 EB      JSR      GWIN
1110: 0240 4C 03      JMP      MAINA
1120: 0243 C9 0D      MAING  CMPIM $0D      'D'
1130: 0245 D0 06      BNE      MAINH
1140: 0247 20 D5      JSR      GPLUS
1150: 024A 4C 03      JMP      MAINA
1160: 024D C9 0A      MAINH  CMPIM $0A      'A'
1170: 024F D0 06      BNE      MAINI
1180: 0251 20 22      JSR      TIEN
1190: 0254 4C 03      JMP      MAINA
1200: 0257 C9 13      MAINI  CMPIM $13      'GO'
1210: 0259 D0 06      BNE      MAINJ
1220: 025B 20 4A      JSR      REKEN
1230: 025E 4C 03      JMP      MAINA
1240: 0261 20 6B      JSR      SHIFT
1250: 0264 05 FA      ORA      POINTL
1260: 0266 85 FA      STA      POINTL
1270: 0268 4C 03      JMP      MAINA
1280:
1290: 026B A0 04      SHIFT  LDYIM $04
1300: 026D 06 4B      SHIFTA ASL      TEMPB
1310: 026F 26 FA      ROL      POINTL
1320: 0271 26 FB      ROL      POINTH
1330: 0273 88      DEY
1340: 0274 D0 F7      BNE      SHIFTA
1350: 0276 60      RTS
1360:
1370: 0277 A5 FA      STORE  LDA      POINTL
1380: 0279 95 50      STAAX  TABEL
1390: 027B A5 FB      LDA      POINTH
1400: 027D 95 51      STAAX  TABEL +01
1410: 027F 60      RTS
1420:
1430: 0280 B5 50      SHOW  LDAAX  TABEL
1440: 0282 85 FA      STA      POINTL
1450: 0284 B5 51      LDAAX  TABEL +01
1460: 0286 85 FB      STA      POINTH
1470: 0288 60      RTS
1480:
1490: 0289 A9 00      CLEAR  LDAM $00
1500: 028B 85 FA      STA      POINTL

```

1510:	028D	85	FB	STA	POINTH	
1520:	028F	85	4B	STA	TEMPB	
1530:	0291	60		RTS		
1540:						
1550:	0292	85	FA	SETCYF	LDA	POINTL
1560:	0294	F0	14	BEO	SETCYC	de 2e plaats van links; behalve
1570:	0296	C9	10	CMPI	\$10	bij de 10.
1580:	0298	F0	0A	BEO	SETCYB	
1590:	029A	85	FB	SETCYA	LDA	POINTH
1600:	029C	D0	0C	BNE	SETCYC	
1610:	029E	20	6B	02	SETCYC	
1620:	02A1	4C	9A	02	JMP	SETCYA
1630:	02A4	20	6B	02	SETCYB	SETCYA
1640:	02A7	20	6B	02	SETCYC	SETCYB
1650:	02AA	60		RTS		
1660:						
1670:	02AB	20	92	02	PLUS	SETCYF
1680:	02AE	20	77	02	SETCYF	STORE
1690:	02B1	E8			INX	
1700:	02B2	E8			INX	
1710:	02B3	85	F9	LDA	INH	
1720:	02B5	18		CLC		
1730:	02B6	F8		SED		
1740:	02B7	69	01	ADCI	\$01	
1750:	02B9	D8		CLD		
1760:	02BA	85	F9	STA	INH	
1770:	02BC	20	80	02	SHOW	
1780:	02BF	60		RTS		
1790:						
1800:	02C0	20	92	02	MIN	SETCYF
1810:	02C3	20	77	02	SETCYF	STORE
1820:	02C6	CA			DEX	
1830:	02C7	CA			DEX	
1840:	02C8	85	F9	LDA	INH	
1850:	02CA	38		SEC		
1860:	02CB	F8		SED		
1870:	02CC	E9	01	SBCI	\$01	
1880:	02CE	D8		CLD		
1890:	02CF	85	F9	STA	INH	
1900:	02D1	20	80	02	SHOW	
1910:	02D4	60		RTS		
1920:						
1930:	02D5	20	92	02	PLUS	SETCYF
1940:	02D8	18		CLC		
1950:	02D9	F8		SED		
1960:	02DA	85	FA	LDA	POINTL	
1970:	02DC	69	25	ADCI	\$25	
1980:	02DE	85	FA	STA	POINTL	
1990:	02E0	85	FB	LDA	POINTH	
2000:	02E2	69	00	ADCI	\$00	

2010:	02E4	85	FB	STA	POINTH
2020:	02E6	D8		CLD	
2030:	02E7	20	77 02	JSR	STORE
2040:	02E9	60		RTS	
2050:					
2060:	02EB	20	92 02	JSR	SETCYF
2070:	02EE	38		SEC	
2080:	02EF	F8		SED	
2090:	02F0	A5	FA	LDA	POINTL
2100:	02F2	E9	25	SBCIM	#25
2110:	02F4	85	FA	STA	POINTL
2120:	02F6	A5	FB	LDA	POINTH
2130:	02F8	E9	00	SBCIM	#00
2140:	02FA	85	FB	STA	POINTH
2150:	02FC	D8		CLD	
2160:	02FD	20	77 02	JSR	STORE
2170:	0300	60		RTS	
2180:					
2190:	0301	20	89 02	JSR	CLEAR
2200:	0304	A4		TAX	
2210:	0305	85	49	STA	KEY
2220:	0307	95	50	STAX	TABEL
2230:	0309	E8		INX	
2240:	030A	E0	50	CPXIM	#50
2250:	030C	D0	F9	BNE	INITA
2260:	030E	A9	01	LDAIM	#01
2270:	0310	85	F9	STA	INH
2280:	0312	A2	00	LDXIM	#00
2290:	0314	60		RTS	
2300:					
2310:	0315	A5	FB	LDA	POINTH
2320:	0317	29	0F	ANDIM	#0F
2330:	0319	D0	06	BNE	LINKSA
2340:	031B	20	68 02	JSR	SHIFT
2350:	031E	4C	15 03	JMP	LINKS
2360:	0321	60		RTS	
2370:					
2380:	0322	A9	10	LDAIM	#10
2390:	0324	85	FA	STA	POINTL
2400:	0326	60		RTS	
2410:					
2420:	0327	20	92 02	JSR	SETCYF
2430:	032A	20	77 02	JSR	STORE
2440:	032D	A2	00	LDXIM	#00
2450:	032F	A9	01	LDAIM	#01
2460:	0331	85	F9	STA	INH
2470:	0333	20	80 02	JSR	SHOW
2480:	0336	60		RTS	
2490:					
2500:					

2510:	2520:	0337	20	8E	1D	KEYDIS	JSR	SCANDS	
2530:	033A	D0	FB			BNE	KEYDIS	SCANDS	
2540:	033C	20	8E	1D		KEYDIA	JSR	SCANDS	
2550:	033F	F0	FB			BEO	KEYDIA	SCANDS	
2560:	0341	20	8E	1D		JSR	SCANDS		
2570:	034A	F0	FB			BEO	KEYDIA		
2580:	034E	20	F9	1D		JSR	GETKEY		
2590:	0349	60				RTS			
2600:									
2610:	034A	20	92	02		JSR	REKEN	SETCVF	De hierna volgende instructies
2620:	034D	20	77	02		JSR	STORE		Zorgen voor de optelling.
2630:	0350	20	89	02		JSR	CLEAR		
2640:	0353	85	F9			STA	INH		
2650:	0355	AA				TAX			
2660:	0356	AB				TAY			
2670:									
2680:	0357	B9	00	00		REKENA	LDAAY	ZWAAR	De zwaarte
2690:	035A	85	4A			STA	TEMPA		
2700:	035C	F0	32			BEO	REKENE		Als zwaarte 0 is, zijn we klaar.
2710:	035E	B5	50			LDAAX	TABEL		Als het getal 0 is, moeten we
2720:	0360	D0	07			BNE	REKENB		naar het volgende getal, zonder
2730:	0362	B5	51			LDAAX	TABEL		+01 dat de zwaartes worden
2740:	0364	D0	03			BNE	REKENB		opgesteld.
2750:	0366	4C	8A	03		JMP	REKEND		
2760:									
2770:	0369	AB	4A			REKENB	LDA	TEMPA	De zwaartes worden opgesteld
2780:	036B	18				CLC			antwoord in INH
2790:	036C	F8				SED			
2800:	036D	65	F9			ADC	INH		
2810:	036F	D8				CLD			
2820:	0370	85	F9			STA	INH		
2830:									
2840:	0372	F8				REKENC	SED		Optellen van de cijfers, antwoord
2850:	0373	AB	4B			LDA	TEMPB		in POINTH, POINTL, TEMPB
2860:	0375	75	50			ADCA	TABEL		
2870:	0377	85	4B			STA	TEMPB		
2880:	0379	AB	FA			LDA	POINTL		
2890:	037B	75	51			ADCA	TABEL		+01
2900:	037D	85	FA			STA	POINTL		
2910:	037F	AB	FB			LDA	POINTH		
2920:	0381	69	00			ADCI	\$00		
2930:	0383	85	FB			STA	POINTH		
2940:	0385	CE	4A			DEC	TEMPA		
2950:	0387	D0	E9			BNE	REKENC		
2960:	0389	D8				CLD			
2970:									
2980:	038A	E8				REKEND	INH		
2990:	038B	E8				INH			
3000:	038C	CB				INY			

Adres	Opdracht	Opmerking
3010:	JMP REKENA	
3020:		
3030:	REKENE JSR LINKS	Het antwoord wordt zo ver
3040:		moeslijk naar links
3050:		geschoven.
3060:		
3070:	LDAIM \$00	De hierna volgende instructies
3080:	STA TEMPA	zorgen voor de deling.
3090:	STA TEMPC	
3100:	STA TEMPD	
3110:	STA TEMPD	
3120:	SED	
3130:		
3140:	LDA REKENF	Er wordt 1 bij het antwoord
3150:	CLC	opgeteld. Antwoord in
3160:	ADCIIM \$01	TEMPD, TEMPC, TEMPA
3170:	STA TEMPA	
3180:	LDA TEMPC	
3190:	ADCIIM \$00	
3200:	STA TEMPC	
3210:	LDA TEMPD	
3220:	ADCIIM \$00	
3230:	STA TEMPD	
3240:	SEC	Er wordt 1 keer de deler (INH)
3250:	LDA TEMPB	afgetrokken van
3260:	SBC INH	POINTH, POINTL, TEMPB
3270:	STA TEMPB	
3280:	LDA POINTL	
3290:	SBCIM \$00	
3300:	STA POINTL	
3310:	LDA POINTH	
3320:	SBCIM \$00	
3330:	STA POINTH	
3340:	BMI REKENG	Bij een getal > 8 slaat hij meteen
3350:	JMP REKENF	daarvan 'links' niet helemaal.
3360:		
3370:	SEC REKENG	Er is 1 keer teveel afgetrokken
3380:	LDA TEMPA	dat wordt gecorrigeerd; bovendien
3390:	SBCIM \$01	wordt het antwoord nu op
3400:	STA TEMPB	POINTH, POINTL, TEMPB gezet
3410:	LDA TEMPC	
3420:	SBCIM \$00	
3430:	STA POINTL	
3440:	LDA TEMPD	
3450:	SBCIM \$00	
3460:	STA POINTH	
3470:	CLD	
3480:	JSR LINKS	
3490:	JSR SHIFT	
3500:	RTS	

Voor het eigenlijke ontwerp te beschrijven wil ik eerst beschrijven hoe mijn KIM is uitgebreid. In het bekende gat van 0400-13FF zit 8K RAM en vanaf 2000 8K RAM. Die 8K RAM bevindt zich voor de helft op de RAM/EPROM kaart uit Elektuur. Eventuele bouwers van die kaart wil ik de volgende tip geven. Velen zullen de 2716 gebruiken. Elektuur schrijft dat de EPROM's in de volgende IC25, IC27, IC26 en IC28 worden geadresseerd, zie Elektuur 203 blz 51 en 52. Dit kan worden opgeheven door niet de draadbruggen R-P en T-S te leggen maar T-P en R-S, van gelsoleerd montage draad. Nu is de volgende IC25, IC26, IC27, IC28, als je naar de tekening op blz 49 zie je waarom dit zo is. Verder hangt aan de inputlijnen PA1-PA7 (7 bits) een ASCII toetsenbord, de strobe daarvan zit met een transistor aan de IRQ. (zie afb.1).

Voor het hier beschreven display zijn nodig natuurlijk een oscilloscoop en twee Digitaal naar Analooq Converters (DAC's) opgebouwd uit 11 weerstanden en 9 buffers, de ene DAC is een 6 bit en de andere een 3 bit DAC. De 6 bit DAC hangt aan de X ingang van de scoop en de 3 bit DAC aan de Y ingang. (afb.2) Let op ! De electronen bundel van mijn scoop wordt naar links afgebogen bij een positieve ingangsspanning. Voor scoop's met een gewone afbuiging kan ipv de 4050 een inverterende 4049 buffer worden gebruikt.

Het is nu de bedoeling door zeer snel wegsturen van verschillende spanningen op de scoop een puntjes display te veroorzaken. Er zijn dan $64 \times 8 = 512$ puntjes die onafhankelijk kunnen worden aangestuurd. Het is duidelijk dat hiervoor 64 bytes moeten worden gereserveerd. Ik heb gekozen voor 0140-017F, ja, midden in de stack. Voor het zichtbaar maken van de puntjes is een zo snel mogelijk programma gemaakt dat net als SCANDS dat vereist, zo vaak mogelijk moet worden aangeroepen.

Voor het display heb ik drie routines geschreven :

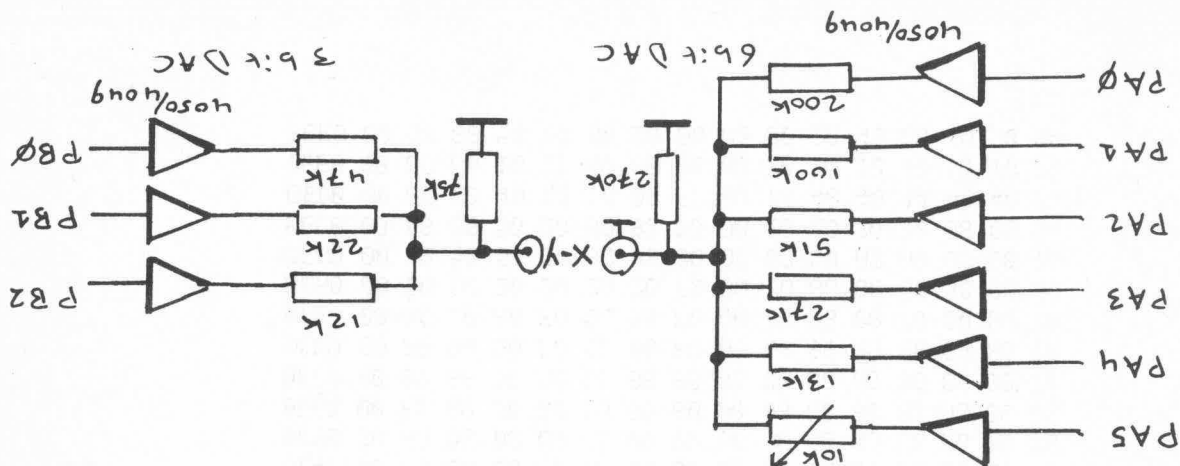
1. DISP . Deze routine laat de in het geheugen opgeslagen punten zien (0140 - 017F)

2. SHIFT. Deze routine gebruikt de ASCII code in A om het adres te berekenen in de tabel waar de punten bij dit ASCII karakter staan, deze worden dan in 0180 - 0187 gezet. Daarna wordt de hele regel (10½ karakter) een punt naar links geschoven, ook de letter in 0180 - 0187 wordt een punt naar links geschoven. Daarna wordt de puntjes buffer gedurende 0,041 s getoond, waarna weer een puntje wordt ingeschoven etc.

Een karakter neemt 6 punten in de breedte in, waarvan het zesde uit is. Door de LDIM \$07 in lokatie \$0F79 te veranderen in LDIM \$07 worden er tussen de karakters twee punten uit gelaten. Dan gaan er nog maar 9 karakters op een regel helaas. Een karakter is 8 punten hoog.

3. SPACE, INPUT en OUT zorgen dat de gewenste letter op het display komt, een Return (\$0D) levert twee spaties op, simpel te veranderen in een of geen spaties.

De INPUT is specifiek voor mijn IRQ routine, die namelijk de ASCII code in een buffer zet. De INPUT routine roept DISP aan en laat de de display buffer zien. In BUFFER (\$1780) staat hoeveel karakters er in de buffer zitten. Als er geen karakters in de buffer staan wordt gewacht op invoer, anders het eerste karakter uit de buffer gehaald, via OUT wordt het karakter getoond en uiteindelijk aan de vragende routine terug gegeven. Essentieel is deze invoer routine niet.



```

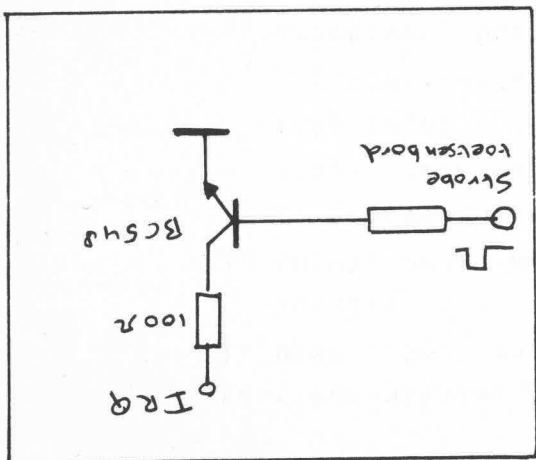
0000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0100 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0120 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0130 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0140 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0150 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0160 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0170 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0180 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0190 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0200 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0210 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0220 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0230 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0240 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0250 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0260 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0270 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0280 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0290 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
02F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0300 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0310 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0320 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0330 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0340 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0350 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0360 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0370 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0380 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0390 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
03F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

Gegeven indeling :
 \$0140 - \$017F te displayen punten
 64x8
 \$0180 - \$0187 nieuwe in te schuiven
 karakter
 \$0D00 - \$0EFF karakter punten tabel
 \$0F00 - \$0FDE routines DISP, SHIFT
 en SPACE, INPUT en OUT

Tabel met punten, 1 letter is 8 bytes, bevat de ASCII codes

\$21 tm \$5F, adres karakter = \$0D00 + ((ASCII-code -\$21) AND \$3F)



```

0010: *****
0020: *
0030: * DISPLAY VAN 64*8 PUNTEN
0040: * OF EEN OSCILLOSCOOP
0050: *
0060: * C. TOTTE 1981
0070: *
0080: *****
0090: ;
0100: ; DEFINITIES :
0110: ;
0120: DISP ORG $0F00
0130: ;
0140: DOPB * $1703 ; V-SPANNING
0150: PB * $1702
0160: DOPB * $1741
0170: PA * $1740
0180: UX * $00E8
0190: DISBVT * $00E8
0200: COUNT * $00EB
0210: ;
0240: 0F00 R9 08
0250: 0F02 85 EB
0260: 0F04 R9 3F
0270: 0F06 00 41 17
0280: 0F09 80 41 17
0290: 0F0C R9 07
0300: 0F0E 00 03 17
0310: 0F11 80 03 17
0320: 0F14 R9 3F
0330: 0F16 85 EB
0340: 0F18 R9 00
0350: 0F1A R8
0360: 0F1B R8
0370: 0F1C R9 40 01
0380: 0F1F 85 EA
0390: 0F21 26 EA
0400: 0F23 90 06
0410: 0F25 R5 EB
0420: 0F27 80 40 17
0430: 0F28 8E 02 17
0440: 0F2D C6 E8
0450: 0F2F 30 06
0460: 0F31 C6 EB
0470: 0F33 F0 07
0480: 0F35 D0 EA
0490: 0F37 E8
0500: 0F38 R9 3F
0510: 0F38 85 EB
0520: 0F3C C8
0530: 0F3D R9 08
0540: 0F3F 85 EB
0550: 0F41 C0 40
0560: 0F43 D0 D7
0570: 0F45 60

START LDARV $0140 ; DISPLAY BUFFER BEGIN
STRZ DISBVT BEWAAR DAT BYTE
ROLZ DISBVT ; SCHUIF BYTE ROND IN CARRY
BCC TEL
LDARZ UX
STR PA ; OF DRC ZETTEN
STX PB
DECZ UX ; 64 TOT 0 AFTELLEN
BMI INCUV
DECC COUNT
BEG INCV
BNE ROTATE
INX INCV
LDARV $3F ; 64*8 DISPLAY
STRZ UX
INX INCV
STRZ UX
INX INCV
LDARV $08
STRZ COUNT
LDARV $40
CPVIN $40
BNE START
RTS ; ZO NIET DAN AFGELOPEN

```

```

0010: *****
0020: *
0030: * SCHUIF EEN LETTER IN HET DISPLAY
0040: *
0050: * C. TOTTE 1981
0060: *
0070: *****
0080: ;
0090: SHIFT ORG $0F50 ;
0100: ;
0110: ; DEFINES
0120: ;
0130: SAVE * $00ED
0140: RCU * $00E9
0150: TIME * $0028
0160: POINTL * $00E8
0170: TIMER * $1747
0180: POINTH * $00EB
0190: DISP * $0F00
0200: COUNT * $00EC
;
0210:
0220: 0F50 85 E9 START STA RCU SAVE ACCU
0230: 0F52 C9 61 CMPIM #61 MAKK KLEINE LETTERS=GROTE LETTERS
0240: 0F54 90 02 BCC CONT
0250: 0F56 E9 20 SBCIM #20
0260: 0F58 38 SEC
0270: 0F59 E9 20 SBCIM #20
0280: 0F5B 30 56 BMI END
0290: 0F5D 29 3F ANDIM #3F
0300: 0F5F 08 ASLA
0310: 0F60 08 ASLA
0320: 0F61 A2 03 LDXIM #03
0330: 0F63 86 EB STXZ POINTH
0340: 0F65 18 CLC
0350: 0F66 69 80 ADCIM #80
0360: 0F68 26 EB ROLZ POINTH
0370: 0F6A 08 ASLA
0380: 0F6B 26 EB ROLZ POINTH
0390: 0F6D 85 EA STAZ POINTL
0400: 0F6F 80 07 LDVIM #07
0410: 0F71 B1 EA LDIV POINTL
0420: 0F73 99 80 01 STARV #0180
DEV
0430: 0F76 88 BPL LOOP
0440: 0F77 10 F8 LDIM #06
0450: 0F79 A9 06 STAZ COUNT
0460: 0F7B 85 EC STAZ COUNT
0470: 0F7D A2 07 LDXIM #07
0480: 0F7F 86 ED STXZ SAVE
0490: 0F81 A9 01 LDIM #01
0500: 0F83 85 EB STAZ POINTH
0510: 0F85 A5 ED LOOFPB LDHZ SAVE
WELKE REGEL ?
0520: 0F87 08 ASLA
0530: 0F88 08 ASLA
0540: 0F89 08 ASLA
0550: 0F8A 18 CLC
0560: 0F8B 69 40 ADCIM #40

```


STAZ POINTL WIJST EEN BYTE UIT DE LINKSE KOLON VAN

LDXZ SAVE
KOLM \$0180
LDVIM \$07
SHIFT 8 BYTES

LDVIM POINTL DOE HET
KOLM

STAIY POINTL
DEV

BPL LOOFA
DECC SAVE

BPL LOOFPB
LDVIM TIME

STH TIMER
JSR DISP

BIT TIMER
BPL LOOFC

DEC COUNT
BNE LOOFP

LDA ACCU
RTS

0570: 0F8D 85 EA
0580: 0F8F 86 ED
0590: 0F91 3E 80 01
0600: 0F94 80 07
0610: 0F96 81 EA

0620: 0F98 2A
0630: 0F99 91 EA
0640: 0F9B 88
0650: 0F9C 10 F8

0660: 0F9E 06 ED
0670: 0FA0 10 E3
0680: 0FA2 89 28
0690: 0FA4 8D 47 17

0700: 0FA7 20 00 0F
0710: 0FAA 2C 47 17
0720: 0FAD 10 F8
0730: 0FAF 06 EC

0740: 0FB1 D0 C8
0750: 0FB3 85 E9
0760: 0FB5 60

0801: *****

0820: *
0830: * I/O MET TOETSENBOORD EN SCOP

0840: *
0850: * C. TOTTE 1981

0860: *****

0880: ;
0890: ; DEFINES

0100: 0FB6 IO
0110: 0FB6 ORG \$0FB6 ;

0120: 0FB6 BUFFER * \$1780
0130: 0FB6 DISP * \$0F00
0140: 0FB6 SHIFT * \$0F50

0150: 0FB6 SPACE LDVIM \$20 ;

0160: 0FB8 4C C9 0F
0170: 0FBB 20 00 0F

0180: 0FBE AE 80 17
0190: 0FC1 F8 F8

0200: 0FC3 CE 80 17
0210: 0FC6 BD 80 17

0220: 0FC9 48
0230: 0FCA C9 00

0240: 0FCC D0 08
0250: 0FCE 20 B6 0F

0260: 0FD1 20 B6 0F
0270: 0FD4 68

0280: 0FD5 60
0290: *****

0300: 0FDE C9 7F
0310: 0FDE F8 FA

0320: 0FDA 20 50 0F
0330: 0FDD 68

0340: 0FDE 60

VERDA CMPIM \$7F IS HET DEL ?

BEG WEG
JSR SHIFT

PLA
RTS

OUT PHA
LDVIM BUFFER HAAL DIE TOETS

DEC BUFFER
BEG INPUT

LDX BUFFER IS ER EEN TOETS INGEDRUKT ?

JSR DISP
BEG INPUT

JSR SPACE LDVIM \$20 ;

WEG
PLA

RTS

Aanpassingen aan de MACRO ASSEMBLER
van Moser
Van Nieuwenhove Koen
Consciencestraat 60 bus 4
B-3000 Leuven
Belgie

Deze aanpassingen zijn bedoeld voor een Junior-computer, uitgerust met een interfacekaart, met minstens 8K ram, lever 16K. Verder is ook nog de elekterminal of een andere seriële terminal vereist. Aan mijn systeem hanst ook nog een parallelprinter (80-kolom). Deze heb ik er aan gekoppeld via de vrije B522 op de interfacekaart. Ik kocht mijn versie aan bij U. Schröder. Deze levert bij de cassette en het handboek nog een aantal bladen bij die uitlesseven over een mogelijke aanpassing. Het is de bedoeling dat dit samen met mijn uitlesgebruikt wordt. Ik zal mijn best doen zo volledig mogelijk te zijn.

Wat betreft echter de Junior-gebruikers die nu nog deze assembler bij U. Schröder zouden willen aankopen moet ik wel opmerken dat ik aan U. Schröder een copy gestuurd heb van een bij mij MERKENDE versie, zodat hij nu misschien wel al een Junior-versie ter beschikking heeft. Wie weet!

```
0030 ;
0040 ;DE INPUT-OUTPUT NAAR DE TERMINAL
0050 ;
0060 JUN/OUT
0070 DELAY
0080 ;
0090 .BA $3FDE
0100 CRT/CR.FND LDA #$1F
0110 JSR DELAY
0120 LDA #$0A
0130 JSR CRT/OUTPUT
0140 LDA #$1E
0150 JSR DELAY
0160 NOP
0170 NOP
0180 RTS
0190 ;IK HEB DIT STUKJE ER ALLEEN BIJGEVOEGD OMDAT
0200 ;DE RTS OP DE MIJ GELEVERDE CASSETTE ER NIET
0210 ;STOND.
0220 CRT/OUTPUT PHA
0230 NOP
0240 NOP
0250 NOP
0260 NOP
0270 NOP
0280 JSR JUN/OUT
0290 PLA
0300 CMP #$0D
0310 BEQ CRT/CR.FND
0320 RTS
3FDE- A9 1F
3FDE- 20 87 3F
3FDB- A9 0A
3FDD- 20 E8 3F
3FE0- A9 1E
3FE2- 20 87 3F
3FE5- EA
3FE6- EA
3FE7- 60
3FE8- 48
3FE9- EA
3FEA- EA
3FEB- EA
3FEC- EA
3FED- EA
3FEE- 20 34 13
3FF1- 68
3FF2- C9 0D
3FF4- F0 E0
3FF6- 60
```

2059 SYSTEM SOFTWARE

6502 SYSTEM SOFTWARE

0860 :DE INITIËLISATIE VAN DE POORTEN VOOR
 0870 :DE CASSETTEINTERFACE DIE ZICH BEVIND
 0880 :OP \$3FB0 TOT \$3FB8 MOET WEGGELATEN
 0890 :WORDEN.
 0900 :WE ZULLEN DIT VERVANGEN DOOR HET
 0910 :VOLGENDE:

0920	:	0930		3FB0- A9 00
0940		0950		3FB2- EA
0960		0970		3FB3- EA
0980		0990		3FB4- EA
1000				3FB5- EA
1010				3FB6- EA
				3FB7- EA
				3FB8- 60

1020 :
 1030 :
 1040 :DE DOOR MIJ GEBRUIKTE PRINTER IS DE
 1050 :OKI MICROLINE-80
 1060 :DEZE IS AANGESLOTEN OP DE VRIJE 6522.
 1070 :IK MAAK GEBRUIK VAN DE "PRINTER WITH
 1080 :AUTO-LINEFEED" ROUTINE VAN SCHRODER.
 1090 :POORT A WORDT GEBRUIKT VOOR PARALLEL
 1100 :DATA-TRANSPORT. POORT B VOOR DE CON-

1110 :TROLE-SIGNALEN. PB0=STROBE
 1120 :
 1130 :
 1140 :
 1150 :
 1160 :
 1170 :
 1180 :
 1190 :
 1200 :HARDCOPY/A PHA
 1210 :JSR PRINT
 1220 :PLA
 1230 :CMP #\$0D
 1240 :BNE TTTT
 1250 :LDA #\$0A
 1260 :TTTT
 1270 :
 1280 :PRINT
 1290 :LDA #\$FF
 1300 :STA \$1803
 1310 :LDA #\$11
 1320 :STA \$1802
 1330 :LDA \$1800
 1340 :ORA #\$01
 1350 :STA \$1800
 1360 :PLA
 1370 :STA \$1801
 1380 :LDA \$1800
 1390 :ORA #\$10
 1400 :STA \$1800

38D4- 20 00 40	1170	: BA \$38D4
	1180	: JSR HARDCOPY/A
	1190	: BA \$4000
	1200	: HARDCOPY/A PHA
4000- 48	1210	
4001- 20 00 40	1220	
4004- 68	1230	
4005- C9 0D	1240	
4007- D0 02	1250	
4009- A9 0A	1260	
400B- 60	1270	
	1280	
400C- 48	1290	
400D- A9 FF	1300	
400F- 8D 03 18	1310	
4012- A9 11	1320	
4014- 8D 02 18	1330	
4017- AD 00 18	1340	
401A- 09 01	1350	
401C- 8D 00 18	1360	
401F- 68	1370	
4020- 8D 01 18	1380	
4023- AD 00 18	1390	
4026- 09 10	1400	
4028- 8D 00 18		


```

402B-AD 00 18 1410 WAIT LDA $1800 AND #$04 BNE WAIT
402E-29 04 1420 AND #$04 BNE WAIT
4030-D0 F9 1430 LDA $1800 AND #$FE AND #$01 ORA #$01 STA $1800 AND $FF
403F-2D 00 1490 AND $FF STA $1800
4042-8D 00 1500 STA $1800 RTS
4045-60 1510

402B-AD 00 18 1410 WAIT LDA $1800 AND #$04 BNE WAIT
402E-29 04 1420 AND #$04 BNE WAIT
4030-D0 F9 1430 LDA $1800 AND #$FE AND #$01 ORA #$01 STA $1800 AND $FF
403F-2D 00 1490 AND $FF STA $1800
4042-8D 00 1500 STA $1800 RTS
4045-60 1510

1520 ;
1530 ;
1540 ;WAT BETREFT DE CASSETTE INPUT-
1550 ;OUTPUT HEB IK ZOVEEL MOGELIJK
1560 ;GEBRUIK GEMAAKT VAN DE IN DE
1570 ;JUNIOR AANWEZIGE SUBROUTINES.
1580 ;
1590
3FA3-20 48 40 JSR LOAD
3FAD-20 48 40 JSR LOAD
1600
1610
1620
1630 JSR RECORD
1640 ;
1650 ;
1660
4048-AD 10 01 1670 LOAD
404B-8D 79 1A 1680 STA $1A79
404E-AD 24 01 1690 LDA $124
4051-8D 70 1A 1700 STA $1A70
4054-AD 25 01 1710 LDA $125
4057-8D 71 1A 1720 STA $1A71
405A-AD 26 01 1730 LDA $126
405D-8D 72 1A 1740 STA $1A72
4060-AD 27 01 1750 LDA $127
4063-8D 73 1A 1760 STA $1A73
4066-20 DF 09 JSR $09DF
4069-20 BC 14 JSR $14BC
406C-60 1790 RTS
4077-8D 83 1A 1950 STA $1A83
4075-A9 7E 1940 LDA #$7E
4072-8D 78 1A 1930 STA $1A78
406F-8D 82 1A 1920 STA $1A82
406D-A9 32 1910 RECORD LDA #$32
1900 ;
1890 ;
1880 ;HEADER MEGGESCHREVEN.
1870 ;ER WORDT NAWELIJK EERST EEN FILE-
1860 ;TWEEMAAL IETS ZIET LEZEN OF SCHRIJVEN.
1850 ;OPMERKEN DAT PER MEGGESCHREVEN FILE MEN
1840 ;ALLES ZOALS BIJ DE JUNIOR. MEL WIL IK
1830 ;NAAR CASSETTE GEBEURT, VISUEEL GEZIEN,
1820 ;ZOMEL VOOR LEZEN VAN, ALS SCHRIJVEN
1810 ;
1800 ;
1790

```

407A- A9 7F	1960	LDA #7F	
407C- 8D 81 1A	1970	STA \$1A81	
407F- A9 00	1980	LDA #00	
4081- 8D 6E 1A	1990	STA \$1A6E	
4084- 8D 6F 1A	2000	STA \$1A6F	
4087- A9 FF	2010	LDA #FF	
4089- 8D 6B 1A	2020	STA \$1A6B	
408C- 20 C2 0B	2030	JSR \$BC2	BBBB
408F- 6E 6B 1A	2040	ROR \$1A6B	
4092- AD 6B 1A	2050	LDA \$1A6B	
4095- 20 E8 0B	2060	JSR \$BEB	
4098- C9 16	2070	CMP #16	
409A- D0 F0	2080	BNE BBBB	
409C- A0 0A	2090	LDY #0A	
409E- 8C 69 1A	2100	STY \$1A69	
40A1- 20 36 0C	2110	JSR \$C36	DDDD
40A4- 20 5D 0C	2120	JSR \$C5D	
40A7- C9 16	2130	CMP #16	
40A9- D0 DC	2140	BNE CCCC	
40AB- CE 69 1A	2150	DEC \$1A69	
40AE- D0 F1	2160	BNE DDDD	
40B0- 20 36 0C	2170	JSR \$C36	GGGG
40B3- 20 5D 0C	2180	JSR \$C5D	
40B6- C9 2A	2190	CMP #2A	
40B8- F0 06	2200	BEO EEEE	
40BA- C9 16	2210	CMP #16	
40BC- F0 F2	2220	BEO GGGG	
40BE- D0 AD	2230	BNE RECORD	
40C0- 20 5D 0C	2240	JSR \$C5D	EEEE
40C3- 20 F3 0B	2250	JSR \$BF3	
40C6- 20 F3 0B	2260	JSR \$BF3	RRRR
40C9- 20 4B 0C	2270	JSR \$C4B	
40CC- AD 24 01	2280	LDA \$124	
40CF- 8D FA 00	2290	STA \$FA	
40D2- 20 F3 0B	2300	JSR \$BF3	
40D5- 20 4B 0C	2310	JSR \$C4B	
40D8- AD 25 01	2320	LDA \$125	
40DB- 8D FB 00	2330	STA \$FB	
40DE- 20 F3 0B	2340	JSR \$BF3	KKKK
40E1- 30 8A	2350	BMI RECORD	
40E3- F0 1D	2360	BEO HHHH	
40E5- 20 4B 0C	2370	JSR \$C4B	
40E8- A0 00	2380	LDY #00	
40EA- A0	2390	TAX	
40EB- AD 23 01	2400	LDA \$123	
40EE- F0 03	2410	BEO 1111	
40F0- 8A	2420	TAX	
40F1- 91 FA	2430	STA (\$FA), Y	
40F3- 8A	2440	TAX	1111
40F4- EE FA 00	2450	INC \$FA	
40F7- D0 03	2460	BNE JJJJ	
40F9- EE FB 00	2470	INC \$FB	
40FC- 20 64 0C	2480	JSR \$C64	JJJJ
40FF- 4C DE 40	2490	JMP KKKK	
4102- 20 F3 0B	2500	JSR \$BF3	HHHH

4105- CD 6E 1A	2510	CMP \$1A6E	
4108- D0 0C	2520	BNE LLL	
410A- 20 F3 08	2530	JSR \$BF3	
410D- CD 6F 1A	2540	CMP \$1A6F	
4110- D0 04	2550	BNE LLL	
4112- 20 BC 14	2560	JSR \$14BC	
4115- 60	2570	RTS	
4116- 4C 6D 40	2580	JMP RECORD	
	2590		
	2600		
	2610	CONTOLE VAN DE CASSETTE-MOTOR DOOR	
	2620	CONTROL/T, ON EN OFF BEVEL.	
	2630		
	2640		
	2650		
332F- AD 82 1A	2670	LDA \$1A82	
3332- 09 20	2680	ORA #00100000	
3334- 8D 82 1A	2690	STA \$1A82	
	2700	RTS	
3338- AD 82 1A	2720	LDA \$1A82	
333B- 09 40	2730	ORA #01000000	
333D- 8D 82 1A	2740	STA \$1A82	
	2750	RTS	
3341- AD 82 1A	2770	LDA \$1A82	
3344- 29 DF	2780	AND #11011111	
3346- 8D 82 1A	2790	STA \$1A82	
	2800	RTS	
334A- AD 82 1A	2820	LDA \$1A82	
334D- 29 BF	2830	AND #10111111	
334F- 8D 82 1A	2840	STA \$1A82	
	2850	RTS	
	2860		
	2870		
	2880		
	2890		
3C73- AD 82 1A	2900	LDA \$1A82	
3C76- 49 40	2910	ORA #01000000	
3C78- 8D 82 1A	2920	STA \$1A82	
	2930		
3C7B- AD 82 1A	2940	LDA \$1A82	
3C7E- 49 20	2950	ORA #00100000	
3C80- 8D 82 1A	2960	STA \$1A82	
	2970		
	2980		
	2990		

.EN

PATCHES OP BASIC

H.J.C. Otten

In het mei 1981 nummer van Byte verscheen een artikel met de naam: 'Faster BASIC for the Ohio Scientific'. Daarin wordt beschreven hoe bij gebrek aan de ROR instructie bij de eerste 6502 CPU in de Microsoft Basic voor de Challengers een veel tijd vragende vervangings instructie reeks daarvoor was toegepast. Ook de KIM en TIM versies van Microsoft Basic waren voorzien van deze zogenaamde ROR macro's. In Byte van september 1981 vond ik een ingezonden brief waar in werd vermeld wat precies de patches op de KIM versie van Microsoft Basic zijn om de veel snellere ROR instructie toe te passen. Hieronder staan deze patches uit Byte herhaald:

Op de volgende lokaties moet komen te staan:

\$37C2 B0 18	BCS \$ \$37DC	\$38C3 66 73	ROR \$73
\$37D1 76 02	ROR \$02, X	\$38C5 66 74	ROR \$74
\$37D3 76 03	ROR \$03, X	\$38C7 66 75	ROR \$75
\$37D5 76 04	ROR \$04, X	\$38C9 66 76	ROR \$76
\$37D7 68	PLA	\$38CB 66 BD	ROR \$BD
\$37D8 6A	ROR A	\$38CD 98	TYA
\$37D9 C8	INY	\$38CE 4A	LSR
\$37DA D0 E8	BNE \$37C4	\$38CF D0 D6	BNE \$38A7
\$37DC 18	CLC	\$38D1 60	RTS
\$37DD 60	RTS		

(\$37DE tot \$3801 en \$38D2 tot \$3903 zijn nu ongebruikt)

Anderen Basic's van Microsoft zoals PET/CBM en Apple hebben deze patch niet nodig maar voor Challenger bezitters is het nuttig als ze een Disc Basic gebruiken het meinummer van Byte te raadplegen. Basic in Rom bezitters kunnen niets doen zonder nieuwe ROM's te kopen.....

Zelf heb ik in de KIM versie de bovenstaande patches ingevoerd en heb een snelheidsstoename van ruim 10 % kunnen waarmaken. Vooral veel berekeningen hebben bij deze patch baat omdat bij de floating point berekeningen deze routines veelvuldig worden aangeroepen.

Programma Lichtshow voor de JUNIOR en de KIM

K.Kikke
Jonkerlaantje 22
1135 TL Edam

Het programma Lichtshow is gebaseerd op een lopend display. Er zijn al inmiddels verschillende versies hiervan gepubliceerd en een ieder weet dus wel hoe het werkt.

Het programma kent twee modes. Door een reeks tekens al dan niet met \$00 te beginnen geeft men aan welke mode gewenst is. De 00-mode moet worden begonnen en afgesloten met \$00. De reeks tekens tussen de nullen worden dan achtereenvolgens gedisplayed als een gewone lopend display. De niet-00-mode geeft aan hoe vaak een teken moet worden gedisplayed. 00 7F betekent bijvoorbeeld 7 maal het streepje (- = 77) laten zien, waarmee 5 geheugenplaatsen worden bespaard. De niet-00-mode bevat achtereenvolgens het aantal en het teken.

De gehele reeks wordt afgesloten met \$FF. Het display loopt dan leeg en via een sprong naar het begin van het programma begint de Lichtshow weer van voren af aan. Door de sprong opdracht te vervangen door een RTS kan het programma als subroutine worden gebruikt. Voordat de Lichtshow wordt aangeroepen moet het begin adres van het te displayen geheugen worden vermeld in \$00FF en \$0010 (MSB)

Als de Lichtshow zelfstandig wordt gebruikt moet met de hand \$000F en \$0010 worden gevuld en het programma op \$0200 worden gestart.

Een voorbeeld van een tekenreeks is het volgende, waarbij het start adres op 0300 moet worden gezet :

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0300	06	77	00	6F	5F	00	06	7E	00	7D	7B	00	04	77	00
0310	7E	3F	77	77	3F	7E	00	0A	3F	07	77	03	37	07	36
0320	37	04	77	00	6F	3F	7D	6F	3F	76	6E	3F	7D	6F	3F
0330	5D	59	00	03	49	04	6B	06	2B	05	67	00	77	77	3F
0340	79	79	7D	7D	1C	54	1C	54	1C	54	1C	54	1C	54	1C
0350	54	00	08	77	FF										

Noot van de redactie voor KIM gebruikers :

Let u op de KIM/JUN definitie, de JUNIOR invertteert niet in de hardware , vandaar dat de software voor de KIM moet invertteren om het zelfde effect als dit oorspronkelijk voor de JUNIOR geschreven programma te geven op de KIM displays. EOR KIM/JUN geeft met 00 geen inversie (JUNIOR) en met FF wel (KIM) . De lichtkrant display code in het geheugen kan hiermee hetzelfde blijven.

; LICHTSHOW OF LED DISPLAYS

; AUTEUR : W.KIKKE

JONKERLAANTJE 22

1135 TL EDAM

LICHT ORG \$0200 ;

; ZERO PAGE VARIABELLEN

; ZP \$0000 ;

BDISPL * ZP ; DISPLAY GEBIED VOOR DIGITS

BD * ZP ;

LCOUNT * ZP ; DISPLAY SNEELHEID

+08 ; V REGISTER OPSLAG

+09 ; INDEX IN DISPLAY GEHEUGEN

+0C ; ANTAL MAAL NIEUWE DATA

+0E ; NIEUWE DATA

+0F ; ADRES VAN TE DISPLAYEN GEHEUGEN

ADDRH * ZP ;

; PIA ADRESSEN :

PIA * \$1800 ; VOOR KIM #1740

PAD * PIA ; PIA A DATA

PADD * PIA ; PIA DATA DIRECTION

PBD * PIA ; PIA B DATA

PBDD * PIA ; PIA B DATA DIRECTION

; CONSTATES

KIMJUN * \$00 ; VOOR KIM #FF ;

; PROGRAMMA LICHTSHOW

START LDIM #FF ; INDEX OF NUL ZETTEN

STA INDEX ;

LDIM #06 ; 6 DIGITS UITZETTEN

CLEAM STRZX BDISPL ; DOOR DISPLAY GEBIED TE VULLEN

; DEX

BPL CLEAN ;

JSR NAD ; HAAL DATA

BEQ NMODE ; ALS OO MODE DAN NARR NMODE

CMPIM #FF ; ALS EINDE DATA DAN

BEQ LDISP ; DISPLAY LATEN LEEGLOPEN

STA NCOUNT ; ANTAL MAAL NIEUWE DATA

JSR NAD ; HAAL DISPLAY DATA

JSR DISPL ; VOEG TOE OF DISPLAY

JMP NMODE ;

JSR NAD ; HAAL DATA

CMPIM #00 ; ALS EINDE OO MODE

BEQ NMODE ; DAN NARR NMODE

PHR ;

LDIM #01 ; EENMAAL NIEUWE DATA

STA NCOUNT ;

PLR ;

DE 6502 KENNER

```
JSR DISPL : VOEG DIOE RAM DISPLAY
JMP NMODE :
```

```
: ROUTINE OM VOLGENDE ADRES EN DATA TE HALEN
```

```
INC INDEX : VOLGENDE DATAVTE
```

```
LDV INDEX : HAAL DATAVTE
```

```
CPYIM $FF : VOLGENDE PAGINA ?
```

```
BNE NP
```

```
INC ADDRH
```

```
PLR
```

```
RTS
```

```
: DISPLAY LEEG LATEN LOPEN
```

```
LDISP LDAIM $06 : LAAT DISPLAY LEEG
```

```
STA NCOUNT : LOOPEN DOOR
```

```
LDAIM $FF : ZES MAAL UIT
```

```
JSR DISPL : TOE TE VOEGEN
```

```
JMP START
```

```
: ROUTINE DISPLAY
```

```
: :
```

```
DISPL STA NDATR : VOEG NIEUWE DATA TOE
```

```
LDR NDATR : RAM DISPLAY GEBIED
```

```
STA BDISPL : :
```

```
LDAIM $10 : DISPLAY SNEELHEID
```

```
STA LCOUNT : :
```

```
LDAIM $7F : :
```

```
STA PADD : :
```

```
LDAIM $08 : :
```

```
LDAIM $06 : :
```

```
STV VSAVE : :
```

```
LDRAV BDISPL : :
```

```
STX PBD : :
```

```
EORIM KIMJUN : KIM/JUNIOR INVERSIE
```

```
STA PBD : ZET OP DISPLAY
```

```
LDAIM $FF : WACHT EVEN MET DIT DIGIT
```

```
DEV : :
```

```
BNE WAIT : :
```

```
INX : VOLGENDE DIGIT
```

```
INX : :
```

```
LDV VSAVE : :
```

```
DEV : VOLGENDE GEHEUGEN
```

```
BPL LOOP1 : ZET DIGIT AF
```

```
STV PADD : VOLGENDE LOOP
```

```
DEC LCOUNT : :
```

```
BNE LOOP0 : :
```

```
LDAIM $05 : MOVE DISPLAY GEBIED
```

```
LDRAV BDISPL : STRZX BD
```

```
DEX : :
```

```
BPL MOVE : EEN DIGIT VOORUIT
```

```
DEC NCOUNT : MAANTAL MALEN NIEUWE DATA ?
```

```
BNE DIS : :
```

```
RTS
```

```
0600: 022C 20 40 02 JSR
```

```
0610: 022F 4C 1F 02 JMP
```

```
0620: :
```

```
0630: :
```

```
0640: :
```

```
0650: 0232 E6 09 INC
```

```
0660: 0234 B4 09 LDV
```

```
0670: 0236 B1 0F LDRAV ADDR
```

```
0680: 0238 48 PHA
```

```
0690: 0239 C0 FF CPYIM $FF
```

```
0700: 023B D0 02 BNE NP
```

```
0710: 023D E6 10 INC ADDRH
```

```
0720: 023F 68 PLR
```

```
0730: 0240 60 RTS
```

```
0740: :
```

```
0750: :
```

```
0760: :
```

```
0770: 0241 B9 06 LDISP
```

```
0780: 0243 85 0C STA NCOUNT
```

```
0790: 0245 B9 FF LDAIM $FF
```

```
0800: 0247 20 40 JSR
```

```
0810: 0249 4C 00 JMP
```

```
0820: :
```

```
0830: :
```

```
0840: :
```

```
0850: 024D 85 0E STA
```

```
0860: 024F 85 0E LDR
```

```
0870: 0251 85 00 STA
```

```
0880: 0253 B9 10 LDAIM $10
```

```
0890: 0255 85 07 STA LCOUNT
```

```
0900: 0257 B9 7F LDAIM $7F
```

```
0910: 0259 80 81 STA PADD
```

```
0920: 025C B2 08 LDAIM $08
```

```
0930: 025E B0 06 LDAIM $06
```

```
0940: 0260 84 08 STV VSAVE
```

```
0950: 0262 B9 00 LDRAV BDISPL
```

```
0960: 0265 8E 82 STX PBD
```

```
0970: 0268 49 00 EORIM KIMJUN
```

```
0980: 026A 80 80 STA PBD
```

```
0990: 026D B0 FF LDAIM $FF
```

```
1000: 026F 88 DEV
```

```
1010: 0270 D0 FD BNE WAIT
```

```
1020: 0272 E8 INX
```

```
1030: 0273 E8 INX
```

```
1040: 0274 B4 08 LDV VSAVE
```

```
1050: 0276 88 DEV
```

```
1060: 0277 10 E7 BPL LOOP1
```

```
1070: 0279 8C 81 STV PADD
```

```
1080: 027C C6 07 DEC LCOUNT
```

```
1090: 027E D0 D7 BNE LOOP0
```

```
1100: :
```

```
1110: 0280 B2 05 LDAIM $05
```

```
1120: 0282 B5 00 LDRAV BDISPL
```

```
1130: 0284 95 01 STRZX BD
```

```
1140: 0286 C8 DEX
```

```
1150: 0287 10 F9 BPL MOVE
```

```
1160: 0289 C6 0C DEC NCOUNT
```

```
1170: 028B D0 C2 BNE DIS
```

```
1180: 028D 60 RTS
```

4 KOLOM printer programma print 4 kolommen met 16 regels op het display van de Elektterminal. Het beginadres van het te displayen stuk moet op \$0000 (LSB) en \$0001 (MSB) worden gezet. Hetprogramma print dan adres , opcode en eventuele operanden. Andere terminals zijn ook bruikbaar mits voorzien van cursor besturing.

0500:	0211	A9	0C	LDIM FF	; CLEAR SCREEN
0510:	0213	20	34	JSR PRCHA	
0520:	0216	A9	80	LDIM #80	
0530:	0218	80	F7	STA TIMER	
0540:	021B	2C	F7	BIT TIMER	
0550:	021E	10	FB	BPL CHECK	
0560:	0220	A9	00	LDIM #00	
0570:	0222	85	03	STA HTEL	
0580:	0224	A6	03	LDX HTEL	; IF HOR POS < > 0 THEN
0590:	0226	F0	08	BEQ HIBYTE	; WHILE CURSOR POS < > HTEL DO
0600:	0228	A9	09	LDIM #09	
0610:	022A	20	34	JSR PRCHA	; CURSOR POS := CURSOR POS + 1
0620:	022D	CA		DEX	
0630:	022E	D0	F8	BNE RIGHT	
0640:	0230	A5	FB	LDH POINTH	; PRINT (POINTER)
0650:	0232	20	8F	JSR PRVTE	
0660:	0235	A5	FB	LDH POINTL	
0670:	0237	20	8F	JSR PRVTE	
0680:	023A	20	F3	JSR PRSP	; PRINT (SPATIE)
0690:	023D	A0	00	LDIM #00	
0700:	023F	B1	FA	LDIV POINTL	
0710:	0241	20	60	JSR LENACC	
0720:	0244	A6	F6	LDX BYTES	
0730:	0246	A0	00	LDIM #00	
0740:	0248	B1	FA	LDIV POINTL	; WHILE INSTR > 0 DO
0750:	024A	20	8F	JSR PRVTE	; PRINT(POINTER.DATA)
0760:	024D	20	F3	JSR PRSP	; PRINT (SPATIE)
0770:	0250	E6	FA	INC POINTL	; POINTER := POINTER + 1
0780:	0252	D0	02	BNE DOWNX	
0790:	0254	E6	FB	INC POINTH	
0800:	0256	CA		DEX	
0810:	0257	D0	EF	BNE PRDATA	
0820:	0259	E6	02	INC VTEL	; VERT POS := VERT POS + 1
0830:	025B	A5	02	LDH VTEL	; IF VERT POS < 16 THEN
0840:	025D	C9	10	CMPIM #10	
0850:	025F	F0	05	BEQ PLUSHO	; PRINT CURR RETURN + LINEFEED
0860:	0261	20	E8	JSR CRLF	
0870:	0264	F0	C2	BEQ RIGHT	; MOVE TO COLUMN
0880:	0266	18		CLC	
0890:	0267	A5	03	LDH HTEL	; HOR POS := HOR POS + 1
0900:	0269	69	0F	ADCM #0F	
0910:	026B	85	03	STA HTEL	
0920:	026D	C9	3C	CMPIM #3C	; IF HOR POS = 64 THEN EXIT
0930:	026F	D0	03	BNE BACK	
0940:	0271	4C	10	JMP MONITO	
0950:	0274	A9	00	LDIM CR	; (SEND CURSOR HOME)
0960:	0276	20	34	JSR PRCHA	; PRINT CURR RETURN
0970:	0279	A2	0F	LDIM #0F	; COUNT := 16
0980:	027B	A9	08	LDIM UP	; WHILE COUNT > 0 DO
0990:	027D	20	34	JSR PRCHA	; PRINT (CURSOR UP)
1000:	0280	CA		DEX	
1010:	0281	D0	F8	BNE UPB	; END WHILE
1020:	0283	F0	9B	BEQ PRINTB	; ENDWHILE (MAINLOOP)


```

0570: 0318 80 02 17 STA PIABD
0580: 031E 20 56 03 JSR TDELAY
0590: 0321 A2 08 LDXIM #08 8 BITS VERSTUREN
0600: 0323 10 00 BPL VOLGB DUMMY
0610: 0325 A0 02 17 LDA PIABD
0620: 0328 46 FE LSR CHAR LSB IN CARRY
0630: 032B B0 05 BCS EEN
0640: 032C 09 01 ORAIM #01
0650: 032E 90 03 BCC BUIT
0660: 0330 EA NOP
0670: 0331 29 FE EEN ANDIM #FE
0680: 0333 80 02 17 BUIT STA PIABD
0690: 0336 20 56 03 JSR TDELAY
0700: 0339 CA DEX
0710: 033A D0 E9 BNE VOLGB
0720: 033C A0 02 17 LDA PIABD
0730: 033F 29 FE ANDIM #FE STOPBIT
0740: 0341 EA NOP
0750: 0342 EA NOP
0760: 0343 EA NOP
0770: 0344 10 00 BPL DUMM
0780: 0346 80 02 17 DUMM STA PIABD
0790: 0349 20 56 03 JSR TDELAY
0800: 034C 68 PLA
0810: 034D 85 FE STA CHAR
0820: 034F 68 PLA
0830: 0350 A8 TRX
0840: 0351 68 PLA
0850: 0352 A8 TRV
0860: 0353 68 PLA
0870: 0354 28 PLP
0880: 0355 60 RTS
0890:
0900:
0910:
0920: 0356 A0 22 TDELAY LDYIM #22
0930: 0358 88 DLOOP DEV
0940: 0359 D0 FD BNE DLOOP
0950: 035B EA NOP
0960: 035C 60 RTS
0970:
0980:
0990:
1000:
1010:
1020:
1030:
1040: 035D 2C 00 17 INKEYB BIT
1050: 0360 10 FB BPL INKEYB BIT 7 IS STROBE
1060: 0362 2C 00 17 STROBE BIT
1070: 0365 30 FB BMI STROBE
1080: 0367 A0 00 17 LDA PIABD
1090: 036A 48 PHA
1100: 036B 20 A0 1E JSR ECHO CHARACTER
1110: 036E 68 PLA
1120: 036F 60 RTS

```

```

1130:
1140: INPUT RAM MICRO ADE HANGEN
1150:
1160: PKB PHA
1170: LDAIM #00
1180: STA PIHADD
1190: LDAIM INKEYB
1200: STA $2E9E MICRO ADE INPUT VECTOR
1210: LDAIM INKEYB /
1220: STA $2E9F
1230: PLA
1240: JMP MAWS
1250:
1260: SERIE KEYBOARD VIA KIM TTY RAM MICRO ADE HANGEN
1270:
1280: SKB LDAIM GETCH
1290: STA $2E9E
1300: LDAIM GETCH /
1310: STA $2E9F
1320: JMP MAWS
1330:
1340: PRINTER PARALLEL VIDEO RAM MICRO ADE HANGEN
1350:
1360: PRAAN LDAIM PIHADD
1370: ORAIM #01
1380: ANDIM #7D
1390: STA PIHADD BIT0 UITGANG, BIT1 INGANG
1400: LDAIM PRAANI
1410: STA $2E91
1420: LDAIM PRAANI /
1430: STA $2E92 VERWANG MICRO ADE OUTPUT VECTOR
1440: LDAIM #00
1450: JSR UITPR
1460: LDAIM #1B ESCAPE NAAR PRINTER
1470: JSR UITPR
1480: LDAIM #75
1490: JSR UITPR
1500: LDAIM #10
1510: JSR UITPR
1520: JMP MAWS
1530:
1540: PRINTER PARALLEL VIDEO OUTPUT
1550:
1560: PRAANI PHA
1570: JSR OUTCH
1580: PLA
1590: JSR UITPR
1600: RTS
1610:
1620: ALLEEN VIDEO OUTPUT M A
1630:
1640: PRUIT LDAIM OUTCH
1650: STA $2E91 MICRO ADE OUTPUT VECTOR NAAR KIM TTY
1660: LDAIM OUTCH /
1670: STA $2E92
1680: JMP $2031

```

Programma BREAK voor Junior en KIM

Frans Mepschen
Eikenlaan 4
9636 CV Zuidbroek (GR.)
tel . 05985 - 2124

Doel van het programma is het inlassen van enkele break opdrachten in een programma, op die brekpoints zorgt BREAK voor een automatische register uitlezing en het vervangen van de break code door een code uit de tabel.

Break start automatisch na het lezen van een break code via de IRQ vector die van te voren moet worden geladen met 00 en 01 (LSB en MSB) Voor de KIM is de IRQ vector op \$17FE en \$17FF te vinden , bij de JUNIOR op \$1A7E en \$1A7F .

Na het aanroepen van BREAK verschijnen op de displays de inhoud van de registers A, X en Y van links naar rechts respectievelijk. Indrukken van de + toets laat de registers S en P zien op de twee linker en twee midden displays , de rechtse displays zijn gedooft (JUNIOR) en 00 (KIM) .

Weer op de + toets drukken laat weer A, X en Y zien. Door de DATA toets in te drukken wordt BREAK verlaten, de break code wordt vervangen door de waarde uit een tabel , op de displays verschijnt het adres van de lokatie waar de BREAK code stond en onder de PC toets staat het begin adres van het programma.

Voor dat BREAK kan worden gebruikt moeten de volgende adressen worden gevuld :

IRQ vector (zie boven)

\$00D0 start adres te testen programma

\$00D1 " " MSB

\$00D2 Counter, laden met 00

Tabel met vervangende codes voor break opdrachten :

\$00D3 Eerste vervangende code voor eerste break opdracht

\$00DD Laatste " "

VOORBEELD :

Hoofdprogramma :

0200	A9 55	LDA \$55
0202	A2 00	LDX \$00
0204	A0 FF	LDY \$00
0206	0A	ASL
0207	CA	DEX
0208	C8	INY
0209	4C ID IC	RESET

De code op de adressen 0206
vervangen door BRK (00)
00D0 = 00 , 00D1 = 02 , 00D2 = 00
00D3 = 0A
Start programma op 0200 G, geeft display
55 00 FF (resp A, X en Y.
+ toets geeft display B4 FD -- (S en SP)
DATA toets geeft display 0206 0A.
PC toets geeft display 0200 A9


```

0030: ; PROGRAMME BREAK
0040: ;
0050: BREAK ORG $0200
0060: ;
0070: ; ZERO PAGE GEBRUIK :
0080: ;
0090: $0000: STHL * ;
0100: $0001: STHH * ;
0110: $0002: COUNT * ;
0120: $0003: LABEL * ;
0130: $0004: MEML * ;
0140: $0005: MEMH * ;
0150: $0006: SAVER * ;
0160: $0007: PCL * ;
0170: $0008: PCH * ;
0180: $0009: SAVERP * ;
0190: $000A: SAVER * ;
0200: $000B: LAST * ;
0210: $000C: SAVER * ;
0220: $000D: SAVER * ;
0230: $000E: NRDIG * ;
0240: $000F: INH * ;
0250: $0010: POINTL * ;
0260: $0011: POINTH * ;
0270: ;
0280: ; JUNIOR MONITOR ROUTINES :
0290: ;
0300: $0012: SCANDS * ;
0310: $0013: GETKEY * ;
0320: $0014: RESET * ;
0330: ;
0340: ; VOOR DE KIM VERSIE WORDEN DEZE LOKALITIES :
0350: ;
0360: SCANDS * $1F1F ;
0370: GETKEY * $1F6A ;
0380: RESET * $1C22 ;
0390: ;
0400: ; CONSTANTS :
0410: ;
0420: PLUS * $12 ;
0430: DATA * $11 ;
0440: ;
0450: ; BREAK ROUTINE
0460: ;
0470: $0200: $05: BREAKR STH SAVER ; SAVE REGISTERS
0480: $0202: $04: STV SAVER ;
0490: $0204: $06: STX SAVER ;
0500: $0206: $08: PLA ;
0510: $0207: $09: STH SAVER ;
0520: $0209: $0A: TSX ;
0530: $020B: $0C: STX SAVER ;
0540: ;
0550: ; DISPLAY MAIN LOOP
0560: ;
0570: $020C: $05: DISP LDA SAVER ; SHOW R,X,Y ON DISPLAY
0580: $020E: $05: STH POINTH ;
0590: $0210: $05: LDA SAVER ; PLACE R,X,Y IN DISPLAY AREA

```

```

0600: 0212 85 F8 STA POINTL ;
0610: 0214 85 F4 LDA SAVEY ;
0620: 0216 85 F9 STA INH ;
0630: 0218 A9 03 LDHIM #03 ; ZINLOOS UOOR KIM MARK ONSCHULDIG
0640: 021A 85 F6 STA NRDIG ; ( 3 DISPLAYS )
0650: 021C 20 8E 10 JSR SCANDS ; SHOW DISPLAY AREA
0660: 021F 20 F9 10 JSR GETKEY ; GET KEY
0670: 0222 C5 F3 CMP LAST ; WHILE OLDKEY < NEWKEY DO
0680: 0224 F0 F6 BEQ DISPA ; GETKEY
0690: 0226 85 F3 STA LAST ;
0700: 0228 C9 12 CMPIM PLUS ; IF KEY = PLUS THEN SHOW S AND P
0710: 022A F0 07 BEQ DISPB ;
0720: 022C C9 11 CMPIM DATA ; IF KEY = DATA THEN RETURN FROM BREAK
0730: 022E F0 26 BEQ RETURN ;
0740: 0230 4C 10 02 JMP DISPA ; IF OTHER KEY KEEP WAITING
0750: 0233 85 F1 LDA SAVEP ; SHOW S AND P
0760: 0235 85 F8 STA POINTH ;
0770: 0237 85 F2 LDA SAVES ; PLACE S AND P IN DISPLAY AREA
0780: 0239 85 F8 STA POINTL ;
0790: 023B A9 02 LDHIM #02 ; ( 2 DISPLAYS )
0800: 023D 85 F6 STA NRDIG ; ZINLOOS UOOR KIM , VERKANG DOOR : 85 F9
0810: 023F 20 8E 10 JSR DISPC ;
0820: 0242 20 F9 10 JSR GETKEY ; GETKEY
0830: 0245 C5 F3 CMP LAST ; WHILE OLDKEY = NEWKEY DO
0840: 0247 F0 F6 BEQ DISPC ; GETKEY
0850: 0249 85 F3 STA LAST ;
0860: 024B C9 12 CMPIM PLUS ; IF KEY = PLUS THEN SHOW H,X,Y
0870: 024D F0 8D BEQ DISP ;
0880: 024F C9 11 CMPIM DATA ; IF KEY = DATA THEN RETURN FROM BREAK
0890: 0251 F0 03 BEQ RETURN ;
0900: 0253 4C 3F 02 JMP DISPC ; IF OTHER KEY KEEP WAITING
0910: ;
0920: ; RETURN FROM BREAK ROUTINE
0930: ; PC KEY SET TO BEGIN ADDRESS
0940: ; REPLACE BREAK CODE WITH REAL CODE FROM LABEL
0950: ;
0960: 0256 68 RETURN PLA ; CALCULATE ADDRESS OF BREAK
0970: 0257 38 SEC ;
0980: 0258 E9 02 SBCIM #02 ;
0990: 025A 85 DE STA MEML ;
1000: 025C 68 PLA ; SAVE IN MEM
1010: 025D E9 00 SBCIM #00 ;
1020: 025F 85 DF STA MEMH ;
1030: 0261 85 D0 LDA STHL ;
1040: 0263 85 EF STA PCL ; SET PC KEY TO BEGIN ADDRESS
1050: 0265 85 D1 LDA STHH ;
1060: 0267 85 F0 STA PCH ;
1070: 0269 86 D2 LDH COUNT ; BREAK COUNT
1080: 026B 85 D3 LDHIZ LABEL ; CODE
1090: 026D E8 INX ;
1100: 026E 86 D2 STX COUNT ; NEXT BREAKPOINT
1110: 0270 A0 00 LDHIM #00 ;
1120: 0272 91 DE STHIV MEML ; RESTORE REAL CODE AT BREAKPOINT
1130: 0274 85 DE LDA MEML ; SET DISPLAY OF BREAK ADDRESS
1140: 0276 85 FA STA POINTL ;
1150: 0278 85 DF LDA MEMH ;
1160: 027A 85 FB STA POINTH ;
1170: 027C 4C 10 1C JMP RESET ; BACK TO MONITOR

```

FRANS SMEEHIJZEN
LIPPEDAL 19
2904 CL CAPELLE AAN DEN IJSSEL
DE KIM-1

PROGRAMMA: HEX-DUMP T.B.V.

AANGEZIEN IK VOOR HET MERKEN OP MIJN KIM-1, AANGESLOTEN
OP EEN VDU (80X24), BEHOEFTE HAD AAN EEN MEER OVERZICHT-
TELIJKE HEXADECIJMALE DUMP DAN DE KIM-1 LEVERT VIA DE 0-
TOETS, HEB IK HET VOLGENDE PROGRAMMA GEMAAKT.
ZDALS HET VOORBEELD MEERGEEFT WORDT MET EEN KOPREGEL GE-
MERKT WELKE DE MOGELIJKHEID BIEDT EENVOUDIGER DE JUISTE
GEHEUGENLOKATIE TE BEPALEN.
VERDER ZIJN DE :18 EN DE CHECKSUM ER AFGELATEN EN ZIJN ER
TUSSEN DE VERSCHILLENDE CODES SPATIES AANGEBRACHT.
OM TUSSENTIJD DE GEDISPLAY-DE DUMP TE KUNNEN BEKIJKEN,
STOPT HET PROGRAMMA WANNEER HET TE DUMPEN ADRES DE XXFF
OVERSCHRIJDT.
HIERNA KAN DAN EEN KEUZE GEMAAKT WORDEN OM OF DOOR TE GAAN
OF OM OPNIEUW TE BEGINNEN VANAF EEN ANDER IN TE TOETSEN
ADRES.
E.E.A. VOLDOET MISSCHIE AAN DE BEHOEFTE VAN ANDERE KIM-1
(JUNIOR) GEBRUIKERS.
N.B. DE DOOR MIJ GEBRUIKTE ASCII-CODE 1A (CLEAR SCREEEN) IS
UITERAARD VOOR IEDERE VDU VERSCHILLENDE.

```
29F8 A2 00 LDX #000 : PRINT "HEX-DUMP V01."
29FA BD 96 2A LDA 2A96,X
29FD 20 A0 1E JSR 1EA0
2A00 E8 INX
2A01 E0 10 CPX #10
2A03 D0 F5 BNE 29FA
2A05 20 70 FA JSR FA70
2A08 20 9D 1F JSR 1F9D
2A0B 85 01 STA 01
2A0D 20 9D 1F JSR 1F9D
2A10 85 00 STA 00
2A12 20 50 2A JSR 2A50
2A15 A5 01 LDA 01
2A17 20 3B 1E JSR 1E3B
2A1A A5 00 LDA 00
2A1C 20 3B 1E JSR 1E3B
2A1F 20 9E 1E JSR 1E9E
2A22 A0 00 LDY #000
2A24 B1 00 LDA (00),Y : DE ADRESSEN 0000 EN 0001 GEVONDEN
2A26 84 03 STY 03 : ADRESSEN
2A28 20 3B 1E JSR 1E3B
2A2B 20 9E 1E JSR 1E9E
2A2E A4 03 LDY 03
29F8 A2 00 LDX #000 : PRINT "HEX-DUMP V01."
29FA BD 96 2A LDA 2A96,X
29FD 20 A0 1E JSR 1EA0
2A00 E8 INX
2A01 E0 10 CPX #10
2A03 D0 F5 BNE 29FA
2A05 20 70 FA JSR FA70
2A08 20 9D 1F JSR 1F9D
2A0B 85 01 STA 01
2A0D 20 9D 1F JSR 1F9D
2A10 85 00 STA 00
2A12 20 50 2A JSR 2A50
2A15 A5 01 LDA 01
2A17 20 3B 1E JSR 1E3B
2A1A A5 00 LDA 00
2A1C 20 3B 1E JSR 1E3B
2A1F 20 9E 1E JSR 1E9E
2A22 A0 00 LDY #000
2A24 B1 00 LDA (00),Y : DE ADRESSEN 0000 EN 0001 GEVONDEN
2A26 84 03 STY 03 : ADRESSEN
2A28 20 3B 1E JSR 1E3B
2A2B 20 9E 1E JSR 1E9E
2A2E A4 03 LDY 03
```

ADR	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
29F8	A2	00	BD	96	2A	20	A0	1E	E8	E0	10	D0	F5	20	70	FA
2A08	20	9D	1F	85	01	20	9D	1F	85	00	20	50	2A	A5	01	20
2A18	3B	1E	A5	00	20	3B	1E	A4	03	C8	C0	10	D0	EF	20	2F
2A28	00	3B	1E	A5	00	20	3B	1E	A4	03	C8	C0	10	D0	EF	20
2A38	A5	00	69	10	85	00	B0	03	4C	15	2A	E6	01	20	5A	1E
2A48	C9	0D	F0	C9	C9	20	F0	A8	A2	00	BD	5E	2A	20	A0	1E
2A58	E8	E0	38	D0	F5	60	1A	41	44	52	20	20	30	30	20	30
2A68	31	20	30	32	20	30	33	20	30	34	20	30	35	20	30	36
2A78	20	30	37	20	30	38	20	30	39	20	30	41	20	30	42	20
2A88	30	43	20	30	44	20	30	45	20	30	46	0D	0A	0A	1A	48
2A98	45	58	2D	44	55	4D	50	20	56	30	31	2E	0D	0A	03	82
2AA8	76	08	FF	02	FE	40	FE	0C	7A	43	B6	8A	51	A4	35	0C
2AB8	DC	80	3F	00	F7	20	FF	00	D6	82	EF	00	EF	02	B7	00
2AC8	95	26	7F	44	B9	22	FF	08	65	28	4B	24	DF	92	F7	08
2AD8	B3	01	FB	07	FE	80	DB	BA	3A	08	FF	84	BF	F8	BF	52
2AE8	FB	02	FB	1A	FF	24	FF	04	7B	FA	DB	32	76	26	BF	8B
2AF8	9D	00	7D	00	FB	08	8B	20	7F	02	5F	6A	2E	A0	FD	8C

AFDruk-Voortbeeld PROGRAMMA HEX-DUMP

HEX-DUMP V01.
STARTADDRESS:

```

2A30 C8 INY
2A31 C0 10 CPY #10
2A33 D0 EF BNE 2A24
2A35 20 2F 1E JSR 1E2F
2A38 A5 00 LDA 00
2A3A 69 10 ADC #10
2A3C 85 00 STA 00
2A3E B0 03 BCS 2A43
2A40 4C 15 2A JMP 2A15
2A43 E6 01 INC 01
2A45 20 5A 1E JSR 1E5A
2A48 C9 0D CMP #0D
2A4A F0 C9 BEQ 2A15
2A4C C9 20 CMP #20
2A4E F0 A8 BEQ 2A15
2A50 A2 00 LDY #00
2A52 BD 5E 2A LDA 2A5E, X
2A55 20 A0 1E JSR 1EA0
2A58 E8 INX
2A59 E0 38 CPY #38
2A5B D0 F5 BNE 2A52
2A5D 60 RTS
2A5E 1A 41 44 52 20 20 30
: TEKST :
30 20 30 31 20 30 32 : ADR 00 01 02 03 04 05 06 07 08 09 0A 0B 0C
20 30 33 20 30 34 20 :
30 35 20 30 36 20 30 :
37 20 30 38 20 30 39 :
20 30 41 20 30 42 20 :
30 43 20 30 44 20 30 :
45 20 30 46 0D 0A 0A :
48 45 58 2D 44 55 : TEKST HEX-DUMP V01.
4D 50 20 56 30 31 2E :
0A 0A
2A30 C8 INY
2A31 C0 10 CPY #10
2A33 D0 EF BNE 2A24
2A35 20 2F 1E JSR 1E2F
2A38 A5 00 LDA 00
2A3A 69 10 ADC #10
2A3C 85 00 STA 00
2A3E B0 03 BCS 2A43
2A40 4C 15 2A JMP 2A15
2A43 E6 01 INC 01
2A45 20 5A 1E JSR 1E5A
2A48 C9 0D CMP #0D
2A4A F0 C9 BEQ 2A15
2A4C C9 20 CMP #20
2A4E F0 A8 BEQ 2A15
2A50 A2 00 LDY #00
2A52 BD 5E 2A LDA 2A5E, X
2A55 20 A0 1E JSR 1EA0
2A58 E8 INX
2A59 E0 38 CPY #38
2A5B D0 F5 BNE 2A52
2A5D 60 RTS
2A5E 1A 41 44 52 20 20 30
: TEKST :
30 20 30 31 20 30 32 : ADR 00 01 02 03 04 05 06 07 08 09 0A 0B 0C
20 30 33 20 30 34 20 :
30 35 20 30 36 20 30 :
37 20 30 38 20 30 39 :
20 30 41 20 30 42 20 :
30 43 20 30 44 20 30 :
45 20 30 46 0D 0A 0A :
48 45 58 2D 44 55 : TEKST HEX-DUMP V01.
4D 50 20 56 30 31 2E :
0A 0A

```

CASSETTEBIBLIOTHEEK

Zoals in 6502 KENNER 16 al is aangekondigd, gaat de redactie de service uitbreiden met een cassette service. Er zijn op het moment twee cassettes leverbaar, opgenomen in KIM + JUNIOR hypertape formaat. Als de source code (alleen Micro Ade nog) aanwezig is dan wordt op de andere zijde van de cassette dat ook opgenomen. In ieder geval is de object (tweemaal) aanwezig . Wat er op de cassette staat en hoe wordt met een begeleidend schrijven duidelijk gemaakt. Gebruik en beschrijving van de programma's is in de 6502 KENNER of KIM KENNER te vinden. S achter de programma naam betekent ook Micro Ade source

KIM cassette nr 1		JUNIOR CASSETTE nr 1	
1. Microchess versnellen	nr 12 S	1. Aut register uitlezing	nr 16 S
2. Supertape	nr 12 S	2. Moonlander	nr 15 S
3. Reactiesnelheidsmeter	nr 12 S	3. Lotto	nr 15 S
4. Verkeerslichten	nr 16 S	4. Browse	nr 16 S
5. Locatie en Replace	nr 18 S	5. Stopwatch	nr 18 S
6. Eprom programmer	nr 5 S	6. Belgische LOTTO	nr 18 S
7. KIM schaakprogramma	nr 11	7. SC/MP cassette afregeling	nr 18 S
8. Schaakopeningen	nr 8	8. Zenuwslag	nr 17 S
9. Talenstudie hulpprog	nr 13	9. Muziekdoos	nr 17 S
10. Mastermind	nr 4	10. Dag naar week omrekening	nr 18 S
11. Target 1 en 6 kolommen	nr 8	11. Dokatimer	nr 17 S

Programma's in de cassette bibliotheek zijn of origineel van onze clubleden of bewerkingen van First Book of KIM programma's. Alle rechten van de programma's blijven aan de auteurs voorbehouden. De kosten van de cassette zijn alleen maar kostendekkend en niet bedoeld om de programma's te kopen. De KIM club is en wordt geen handel .

Kosten per cassette f 12,50
 Bestellen van de cassettes kan door een girobetaalkaart of bankcheque (groene of EURO) en een briefje met het adres en cassette nummer (KIM = 1 , JUNIOR = 2) in een envelop te sturen aan het redactie -secretariaat (Hans Otten,Ottoborrengoed 33, 3871 MJ Hoevelaken) te zenden Levering kan enige weken duren . Andere wijzen van bestellen bestaan niet .

Te koop : ASCII display module van Visser Assembling Electronics.

Teletype compatible 16 regels met 64 karakters, interface met 2 20 mA current loops en video signaal 1 V top-top.

Prijs f 200,- of BF 3000,- incl verzendkosten en handboek. Van Nieuwenhove Koen, Consciencestraat 60 bus 4 B3000 Leuven België

Te koop : 1 penrecorder Leeds Nartrep met doc, 1 anal 1 marker f 85,-

2 micro fiche reader Philips per stuk - 40,-

2 ponsbandlezers 40 kar-s (transp mech + kop, geen el) - 35,-

1 magn card reader en 20 magn kaarten geen electr - 15,-

20 alma reed relais per stuk - 1,50

B. Eenhuizen, Jol 133 1186 SH Amstelveen tel. 020 - 435348

Gevraagd: Wie heeft documentatie of ervaring met een TA21 van Adler

Information Systems, een machine voor het vastleggen van

boekhouding op cassette met een bolkop printer en een 8080 ?

C. Totté, Gr. Albrechtweg 4 3331 HP Zwijndrecht tel 078-127324

Te koop : Maandblad voor Hobby Electronics, ELO en Hobbit

jaargangen 1977, '78, '79 en '80 voor f 20,- per jaargang

Paul van Niekerk Kievitslaan 8 3233 BE Oostvoorne

KIM club bijeenkomsten in 1982 :

16 januari 20 maart 15 mei ... 18 september ... 20 november

6500 gebruikers groep Delft n E-cafe TH :

40 17 -12 - 1981, 21 - 01 - 1982, 25 - 02 1982

582/5/285
0.06
1.90/0

5834259 2059 5834259 2059

5834259 2059 5834259 2059